Predicting Consumer Default[*]

Stefania Albanesi, University of Pittsburgh, NBER and CEPR
Domonkos F. Vamossy, University of Pittsburgh

January 11, 2021

Abstract

We develop a model to predict consumer default based on deep learning. We show that the model consistently outperforms standard credit scoring models, even though it uses the same data. Our model provides favorable credit risk assessment to young borrowers relative to standard credit scoring models, while accurately tracking variations in systemic risk. We argue that these properties can provide valuable insights for the design of policies targeted at reducing consumer default and alleviating its burden on borrowers and lenders, as well as macroprudential regulation.

JEL Codes: C45; D14; E27; E44; G21; G24.

Keywords: Consumer default; credit scores; deep learning; macroprudential policy.

[*] We are grateful to Dokyun Lee, Sera Linardi, Yildiray Yildirim, Albert Zelevev and seminar participants at the UK Financial Conduct Authority, the University of Pittsburgh, the European Central Bank, Baruch College, Notre Dame University, New York University, the Federal Reserve Banks of Atlanta and Cleveland, and Goethe University for useful comments and suggestions. This research was supported in part by the National Science Foundation under Grant No. SES 1824321. This research was also supported in part by the University of Pittsburgh Center for Research Computing through the resources provided. Correspondence to: stefania.albanesi@gmail.com.

1 Introduction

The dramatic growth in household borrowing since the early 1980s has increased the macroeconomic impact of consumer default. A key determinant of this growth in the United States has been the adoption of risk based approach of allocating credit to individual borrowers, where the risk is measured by credit scoring models. Despite the ubiquitous use of credit scores in everyday financial transactions, little is known about the performance of these models in assessing default risk at the individual level. Additionally, since credit scores are ordinal, the distribution of credit scores provides little guidance on the systemic risk associated with household sector borrowing. Finally, given the proprietary nature of these models and the weak reporting requirements imposed by law, consumers, policymakers and even lenders are left without much insight into the factors affecting credit score variation, leading to a lack of transparency and accountability that could adversely affect outcomes on consumer credit markets.

This paper proposes a novel approach to predicting consumer default based on deep learning. Our methodology uses the same information as standard credit scoring models. We rely on deep learning as this methodology is specifically designed for prediction in environments with high dimensional data and complex non-linear patterns of interaction among variables affecting the outcome of interest, for which standard regression approaches perform poorly. We show that our model improves the accuracy of default predictions compared to conventional a credit score while increasing transparency and accountability. It is also able to track variations in systemic risk, and is able to identify the most important factors driving defaults and how they change over time. We show that conventional credit scores misclassify default risk for approximately 30% of consumer borrowers and that low income and young borrowers receive a credit score that is too low relative to their realized default behavior, which is better predicted by our model. Finally, we show that adopting our model can accrue substantial savings to borrowers and lenders.

Understanding the determinants of consumer default and predicting its variation over time and across types of consumers can not only improve the allocation of credit, but also lead to important insights for the design of policies aimed at preventing consumer default or alleviating its effects on borrowers and lenders. They are also critical for macroprudential policies, as they can assist with the assessment of the impact of consumer credit on the fragility of the financial system.

Our proposed model of consumer default uses the same data as conventional credit scoring models, improves on their performance, benefiting both lenders and borrowers, and provides more transparency and accountability. We resort to deep learning, a type of machine learning ideally suited to high dimensional data, such as that available in consumer credit reports.[1] The inputs of our model are features, such as loan balances and number of trades, delinquency information, and attributes related to the length of a borrower's credit history, to produce an individualized estimate that can be interpreted as a probability of default. We target the same default outcome as conventional credit scoring models, namely a 90+ days delinquency in the subsequent 8 quarters. We present a variety of performance metrics suggesting that our model has very strong predictive ability. Accuracy, that is percent of observations correctly classified, is above 86% for all periods in our sample, and the AUC-Score, a commonly used metric in machine learning, is always above 92%. Our model is interpretable, enabling us to identify the factors that are most strongly associated with default at the individual level and for the entire population. We can also examine how these vary over time in response to changing economic conditions.

Our main contribution is a comparison of our model to a conventional credit score. By construction, credit scores only provide an ordinal ranking of consumers based on their default risk, and are not associated to a specific default probability.[2] Yet, it is still possible to compare performance by assessing whether borrowers fall in different points of the credit score distribution compared to our model predictions. We find that our model performs significantly better than conventional credit scores. The AUC score for the credit score, a measure of the ability to differentiate borrowers based on their credit score is approximately 82% and drops during the 2007-2009 crisis, while the AUC score for our model is approximately 93% and stable over time. Perhaps most importantly, the credit score generates large disparities between the implied predicted probability of default and the realized default rate for large groups of customers, particularly at the low end of the credit score distribution. We show that, among Subprime borrowers, who comprise 21% of the population, 17% display default behavior which is consistent with Near Prime borrowers and 15% display default behavior consistent with Deep Subprime. The default rates for Deep Subprime, Subprime and Near Prime borrowers are respectively 95%, 79% and 44%, so this misclassification is large, and it would imply large losses for lenders and borrowers in terms of missed revenues or higher interest rates. By contrast, the discrepancy between predicted and realized default rates for our model is never more than 4 percentage points.

One concern with adopting a model based on machine learning is that a more sophisticated statistical technology might exacerbate disparities in access to credit for certain disadvantaged consumers such as young, low income or minorities (see Fuster et al. (2018)).

[1] For excellent reviews of how machine learning can be applied in economics, see Mullainathan and Spiess (2017) and Athey and Imbens (2019).

[2] Credit scoring companies provide guidelines to lenders on the relation between credit score values and actual probability of defaults for different economic scenarios.

We show that, to the contrary, our model provides a more favorable risk assessment to young and low income borrowers, particularly for those who do not default. This property is a function of the improved performance of our model compared to a conventional credit score. We show that our model total amounts owed are strongly associated with default, whereas credit demand indicators and the length of the credit history is not as important. By contrast, conventional credit scores are less dependent on total amounts owed, while credit demand factors and length of the credit history have a sizable negative impact on the score.

We also examine the ability of our model to capture the evolution of aggregate default risk. Since our data set is nationally representative and we can score all borrowers with a non-empty credit record, the average predicted probability of default in the population based on our model corresponds to an estimate of aggregate default risk. We find that our model tracks the behavior of aggregate default rates remarkably well. It is able to capture the sharp rise in aggregate default rates in the run up and during the 2007-2009 crisis and also captures the inversion point and the subsequent drastic reduction in this variable. With the growth in consumer credit, household balance sheets have become very important for macroeconomic performance. Having an accurate assessment of the financial fragility of the household sector, as captured by the predicted probability of default on consumer credit has become crucially important and can aid in macro prudential regulation, as well as for designing fiscal and monetary policy responses to adverse aggregate economic shocks. This is another advantage of our model compared to credit scores, since the latter only provides an ordinal ranking of consumers with respect to their probability of default. Our model can provide such a ranking but in addition also provides an individual prediction of the default rate which can be aggregated into a systemic measure of default risk for the household sector.

As a final application, we compute the value to borrowers and lenders of using our model. For consumers, the comparison is made relative to the credit score. Specifically, we compute the credit card interest rate savings of being classified according to our model relative to the credit score. Being placed in a higher default risk category substantially increases the interest rates charged on credit cards at origination and increasingly so as more time lapses since origination, whereas being placed in a lower risk category reduces interest rate costs. We choose credit cards as they are a very popular form of unsecured debt, with 74% of consumers holding at least one credit or bank card. In percentage of credit cards balances, average net interest rate expense savings are approximately 5% for low credit score borrowers. These values constitute lower bounds as they do not include the higher fees and more stringent restrictions associated with credit cards targeted to low credit score borrowers and the increased borrowing limits available to higher credit score borrowers. For lenders, we calculated the value added by using our model in comparison to not having a prediction of default risk or having a prediction based on logistic regression. We use logistic regression for this exercise as it is understood to be the main methodology for conventional credit scoring models. Over a loan with a three year amortization period, we find that the gains relative to no forecast are in the order of 60% with a 15% interest rate, while the gains for relative to a model based on logistic regression are approximately 3%. These results suggest that both borrowers and lenders would experience substantial gains from switching to our model.

Our analysis contributes to the literature on consumer default in a variety of ways. We are the first to develop a prediction model of consumer default using credit bureau data that complies with all of the restrictions mandated by U.S. legislation in this area, and we do so using a large and temporally extended panel of data. This enables us to evaluate model performance in a setting that is closer to the one prevailing in the industry and to train and test our model in a variety of different macroeconomic conditions. Previous contributions either focus on particular types of default or use transaction data that is not admissible in conventional credit scoring models. The closest contributions to our work are Khandani, Kim, and Lo (2010), Butaru et al. (2016) and Sirignano, Sadhwani, and Giesecke (2018). Khandani, Kim, and Lo (2010) apply a decision tree approach to forecast credit card delinquencies with data for 2005-2009. They estimate cost savings of cutting credit lines based on their forecasts and calculate implied time series patterns of estimated delinquency rates. Butaru et al. (2016) apply machine learning techniques to combined consumer trade line, credit bureau, and macroeconomic variables for 2009-2013 to predict delinquency. They find substantial heterogeneity in risk factors, sensitivities, and predictability of delinquency across lenders, implying that no single model applies to all institutions in their data. Sirignano, Sadhwani, and Giesecke (2018) examine over 120 million mortgages between 1995 to 2014 to develop prediction models of multiple states, such as probabilities of prepayment, foreclosure and various types of delinquency. They use loan level and zip code level aggregate information, and provide a review of the literature using machine learning and deep learning in financial economics. Kvamme et al. (2018) predict mortgage default using use convolutional neural networks and emphasize the advantages of deep learning, but they do not evaluate their models out of sample the way we do. Finally, Lessmann et al. (2015) reviews the recent literature on credit scoring, which is based on substantially smaller datasets than the one we have access to, and recommends random forests as a possible benchmark. However, we find that our hybrid model as well as our model components, a deep neural network and gradient boosted trees, improves substantially over random forests, possibly owing to recent methodological advances in deep learning, including the use of dropout, the introduction of new activation functions and the ability to train larger models.[3]

Our model is interpretable, which implies that we are able to assess the most important factors associated with default behavior and how they vary over time. This information is important for lenders, and can be used to comply with legislation that requires lenders and credit score providers to notify borrowers of the most important factors affecting their credit score. Additionally, it can be used to formulate economic models of consumer default. The literature on consumer default[4] suggests that the determinants of default are related to preferences, such as impatience which increases the propensity to borrow, or adverse expenditure of income shocks. Based on these theories, it is then possible to construct theoretical models of credit scoring, of which Chatterjee, Corbae, and Rios-Rull (2011) is a leading example. We find that the number of trades and the balance on outstanding loans are the most important factors associated with an increase in the probability of default, in addition to outstanding delinquencies and length of the credit history. This information can be used to improve models of consumer default risk and enhance their ability to be used for policy analysis and design.

We also identify and quantify a variety of limitations of conventional credit scoring models, particularly their tendency to misclassify borrowers by default risk, especially for relatively risky borrowers. This implies that our default predictions could help improve the allocation of credit in a way that benefits both lenders, in the form of lower losses, and borrowers, in the form of lower interest rates. Our results also speak to the perils associated with using conventional credit scores outside on the consumer credit sphere. As it is well known, credit scores are used to screen job applicants, in insurance applications, and a variety of additional settings. Economic theory would suggest that this is helpful, as long as credit score provide information which is correlated with characteristics that are of interest for the party using the score (Corbae and Glover (2018)). However, as we show, conventional credit scores misclassify borrowers by a very large degree based on their default risk, which implies that they may not be accurate and may not include appropriate information or use adequate methodologies. The broadening use of credit scores would amplify the impact of these limitations.

The paper is structured as follows. Section 2 describes our data. Section 3 discusses the patterns of consumer default that motivate our adoption of deep learning. Section 4 describes our prediction problem and our model. Section 5 compares our model to conventional credit

[3] Other machine learning applications and reviews of default predictions include Moscatelli et al. (2020), Barbaglia, Manzan, and Tosetti (2020), Branzoli and Supino (2020). For deep learning applications see Vamossy (2020) and Sirignano, Sadhwani, and Giesecke (2018).

[4] Some notable contributions include Chatterjee et al. (2007), Livshits, MacGee, and Tertilt (2007), and Athreya, Tam, and Young (2012).

scores. Section 6 illustrates our model's performance in predicting and quantifying aggregate default risk and calculates the value added of adopting our model over alternatives for lenders and borrowers.

2 Data

We use anonymized credit file data from the Experian credit bureau. The data is quarterly, it starts in 2004Q1 and ends in 2015Q4. The data comprises over 200 variables for an anonymized panel of 1 million households. The panel is nationally representative, constructed from a random draw for the universe of borrowers with an Experian credit report. The attributes available comprise information on credit cards, bank cards, other revolving credit, auto loans, installment loans, business loans, first and second mortgages, home equity lines of credit, student loans and collections. There is information on the number of trades for each type of loan, the outstanding balance and available credit, the monthly payment, and whether any of the accounts are delinquent, specifically 30, 60, 90, 180 days past due, derogatory or charged off. All balances are adjusted for joint accounts to avoid double counting. Additionally, we have the number of hard inquiries by type of product, and public record items, such as bankruptcy by chapter, foreclosure and liens and court judgments. For each quarter in the sample, we also have each borrowers's credit score. The data also includes an estimate of individual and household labor income based on IRS data. Because this is data drawn from credit reports, we do not know gender, marital status or any other demographic characteristic, though we do know a borrower's address at the zip code level. We also do not have any information on asset holdings.

Table 1 reports basic demographic information on our sample, including age, household income, credit score and incidence of default, which here is defined as the fraction of households who report a 90 or more days past due delinquency on any trade. This will be our baseline definition of default, as this is the outcome targeted by credit scoring models. Approximately 34% of consumers display such a delinquency.

Table 1: Descriptive Statistics

Feature	Mean	Std. Dev.	Min	25%	50%	75%	Max
Age	45.8	16.3	18	32.2	45.1	57.8	83
Household Income	77.1	55.0	15	42	64	90	325
Credit Score	678.4	111.0	300	588	692	780	839
Default within 8Q	0.339	0.473	0	0	0	1	1

Notes: Credit score corresponds to Vantage Score 3. Household income is in USD thousands, trimmed at the 99th percentile. Source: Authors' calculations based on Experian Data.

3 Patterns in Consumer Default

We now illustrate the complexity of the relation between the various factors that are considered important drivers of consumer default. Our point of departure are standard credit scoring models. While these models are proprietary, the Fair Credit Reporting Act of 1970 and the Equal Opportunity in Credit Access Act of 1984 mandate that the 4 most important factors determining the credit scores be disclosed, together with their importance in determining variation in credit scores. These include credit utilization and number of hard inquiries, which are supposed to capture a consumer's demand for credit, the variety of debt products, which capture the consumer's experience in managing credit, and the number and severity of delinquencies. Each of these factors is stated to account for 10-35% of the variation in credit scores. The length of the credit history is also seen as a proxy on a consumer's experience in managing credit, and this is reported as accounting for 15% of the variation in credit scores.[5] The models used to determine credit scores as a function of these attributes are not disclosed, but they are widely believed to be based on linear and logistic regression as well as score cards. Additionally, available credit scoring algorithms typically do not score all borrowers.

Subsequently, we illustrate the properties of consumer default that suggest deep learning might be a good candidate for developing a prediction model. Specifically, we show that default is a relatively rare but very persistent outcome, there are substantial non-linearities in the relation between default and plausible covariates, as well as high order interactions between covariates and default outcomes.

3.1 Default Transitions

The default outcome we consider is a 90+ days delinquency, which occurs if the borrower has missed scheduled payments on any product for 90 days or more.[6] This is the default outcome targeted by the most widely used credit scoring models, which rank consumers based on their probability of becoming 90+ days delinquent in the subsequent 8 quarters. We refer to borrowers who are either current or up to 60 days delinquent on their payments as current.

The transition matrix from current to 90+ days past due in the subsequent 8 quarters is given in Table 2. Clearly, the two states are both highly persistent, with a 77% of current

[5] For an overview of the information available to borrowers about the determinants for their credit score, see `https://www.myfico.com/resources/credit-education/whats-in-your-credit-score`.

[6] For instance, if no payment has been made by the last day of the month within the past three months and the payment was due on the first day of the month three months ago. For credit cards, this occurs if the borrower does not make at least their minimum payment.

customers remaining current in the next 8 quarters, and 93% of customers in default remaining in that state over the same time period. The probability of transition from current to default is 23%, while the probability of curing a delinquency with a transition from default to current is only 7%. These results suggest that default is a particularly persistent state, and predicting a transition into default is very valuable form the lender's perspective, since they are unlikely to be able to recuperate their losses. But it is also quite difficult, as the current state is also very persistent.

Table 2: Default Transitions

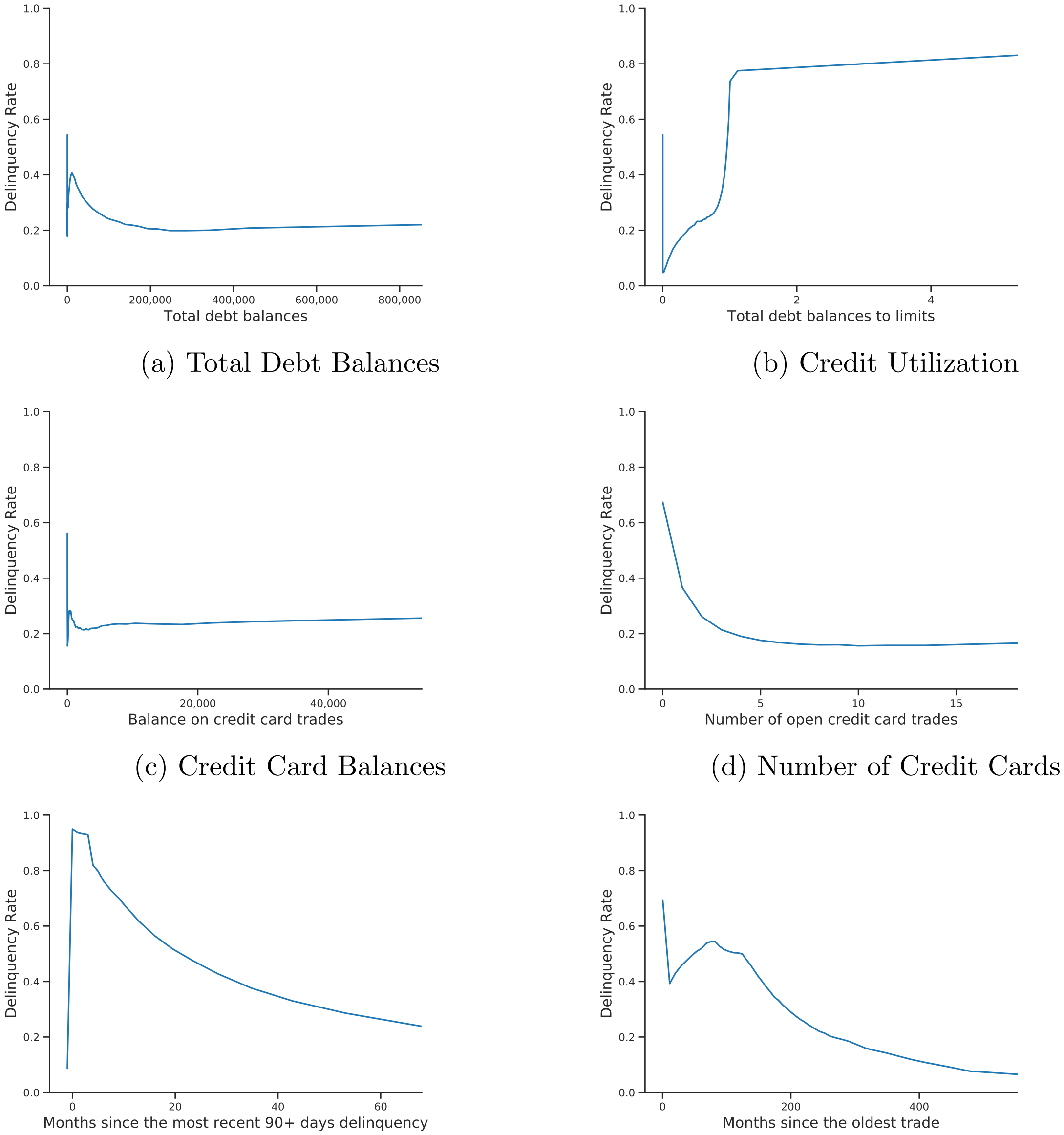
Current/Next 8Q	No default	Default
No default	0.776	0.224
Default	0.073	0.927

Notes: Quarterly frequency of transition from current to default. Current corresponds to 0-89 day past due on any trade, Default corresponds to 90+ day past due on any trade in the subsequent 8 quarters. Source: Authors' calculations based on Experian Data.

3.2 Non-linearities

Our model includes a relatively large list of features, which is presented in Table 9. The summary statistics for these features are reported in Table 10 in Section A.2. As is demonstrated in the table, there is a wide dispersion in the distribution of these variables. For example, the average balance on credit and bankcard trades is approximately $8,800, but the standard deviation, at $19,284, is more than twice as large. Similarly, average total debt balances are approximately $77,000, while the standard deviation is $170,000 and the 75th percentile $95,000, suggesting a high upper tail dispersion of this variable. Other features display similar patterns.

Figure 1 illustrates the highly non-linear relation between selected features and the incidence of default. In particular, it shows how the default rate, defined as the fraction of borrowers with a 90+ day past due delinquency in the subsequent 8 quarters, varies with total debt balances, credit utilization, the credit limit on credit cards, the number of open credit card trades, the number of months since the most recent 90+ day past due delinquency and the months since the oldest trade was opened. The figures show that while the relation between the features and the incidence of default is mostly monotone, it is highly nonlinear, with vary little variation in the incidence of default for most intermediate values of the variable and much higher or lower values at the extremes of the range of each covariate. The variables in the figure are just illustrative, a similar pattern holds for most plausible features.



(a) Total Debt Balances

(b) Credit Utilization

(c) Credit Card Balances

(d) Number of Credit Cards

(e) Proximity to Delinquency

(f) Length of Credit History

Figure 1: Nonlinear Relation Between Default and Covariates

Notes: Delinquency rate is the fraction with 90+ days past due trades in subsequent 8 quarters. In panel (e) and (f), -1 implies no past delinquency. Source: Authors' calculations based on Experian Data.

3.3 High Order Interactions

Multidimensional interactions are another feature of the relation between default and plausible covariates, that is default behavior is simultaneously related with multiple variables. To see this, Figure 2 presents contour plots of the relation between the incidence of default and couples of covariates. The covariates reported here are chosen since they are important driving factors in default decisions, based on our model, as discussed in Section A.4.

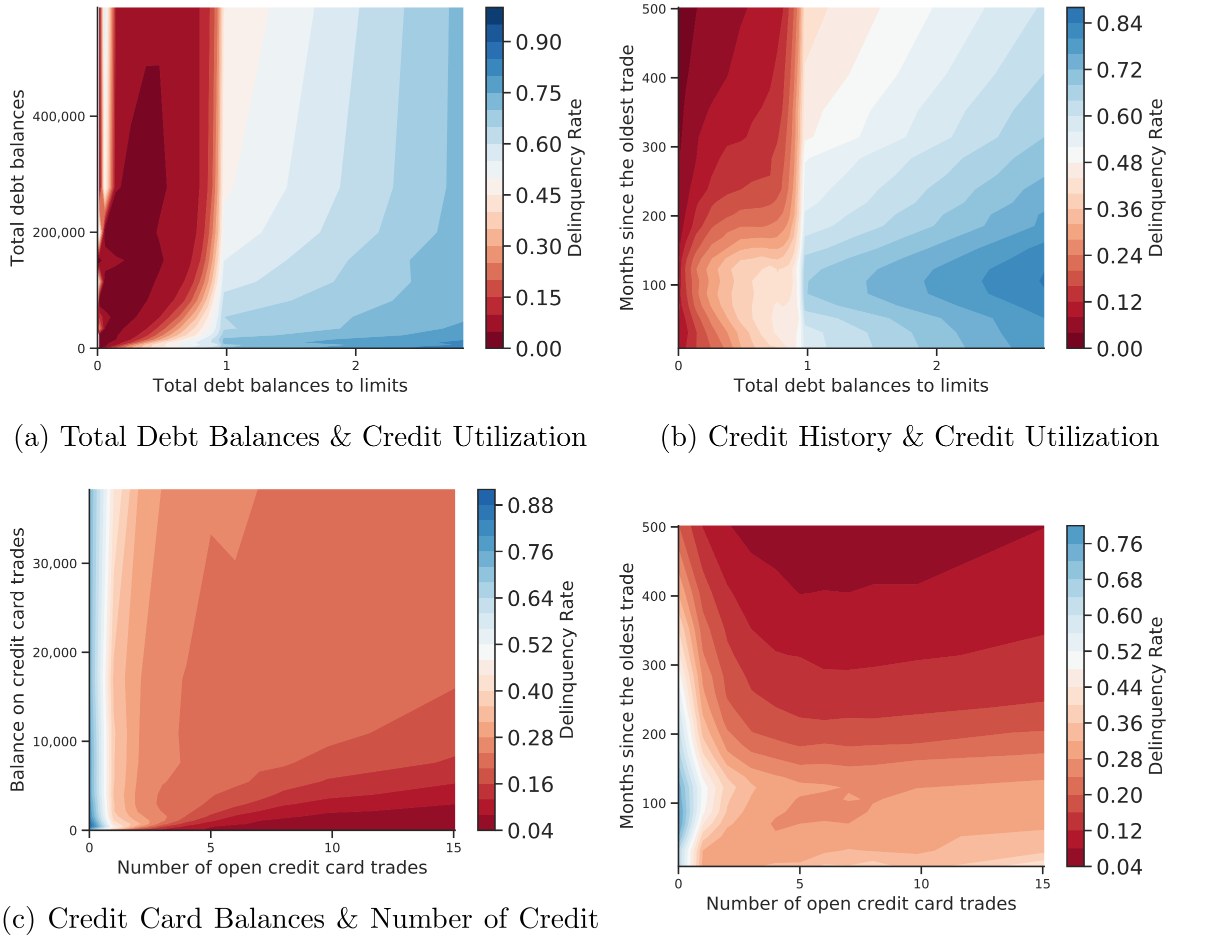
Panels (a) and (b) explore the joint variation in the incidence of default with total debt balances, credit utilization (total debt balances to limits), and credit history. Blue values correspond to high delinquency rates while red values to low delinquency rates. As can be seen from both panels, higher credit utilization corresponds to higher delinquency rate, but for given credit utilization, an increase in total debt balances first decreases then increases the delinquency rate, where the switch in sign depends on the utilization rate. For given utilization rates, a longer credit history first increases then decreases the delinquency rate, provided the utilization rate is smaller than 1.[7] Panels (c) and (d) explore the relation between default and credit card borrowing. Default rates decline with the number of credit cards, though for a given number of credit card trades, they mostly increase with credit card balances. This relation, however varies with the level of both variables. An increase in the length of credit history is typically associated with lower default rates, however, if the number of open credit cards is low, this relation is non-monotone. The variables reported in the figures are illustrative of a general pattern in the joint relation between couples of covariates and default rates.

This pattern of multidimensional non-linear interactions across covariates is fairly difficult to model using standard econometric approaches. For this reason, we propose a deep learning approach to be explained below.

4 Model

Predicting consumer default maps well into a supervised learning framework, which is one of the most widely used techniques in the machine learning literature. In supervised learning, a learner takes in pairs of input/output data. The input data, which is typically a vector, represent pre-identified attributes, also known as features, that are used to determine the output value. Depending on the learning algorithm, the input data can contain continuous and/or discrete values with or without missing data. The supervised learning problem is referred to as a "regression problem" when the output is continuous, and as a "classification

[7] Utilization rates above 1 can arise for a delinquent borrower if fees and other penalty add to their balances for given credit limits.



(a) Total Debt Balances & Credit Utilization

(b) Credit History & Credit Utilization

(c) Credit Card Balances & Number of Credit Cards

(d) Credit History & Number of Credit Cards

Figure 2: Multidimensional Relation Between Default and Covariates

Notes: Relationship between 90+ days past due delinquency rate and pairs of covariates. Source: Authors' calculations based on Experian Data.

problem" when the output is discrete. Once the learner is presented with input/output data, its task is to find a function that maps the input vectors to the output values. A brute force way of solving this task is to memorize all previous values of input/output pairs. Though this perfectly maps the input data to the output values in the training data set, it is unlikely to succeed in forecasting the output values if (1) the input values are different from the ones in the training data set or (2) when the training data set contains noise. Consequently, the goal of supervised learning is to find a function that generalizes beyond the training set, so that it correctly forecasts out-of-sample outcomes. Adopting this machine-learning methodology, we build a model that predicts defaults for individual consumers. We define default as a 90+ days delinquency on any debt in the subsequent 8 quarters, which is the outcome targeted by conventional credit scoring models. Our model outputs a continuous variable between 0 and 1 that can be interpreted under certain conditions as an estimate of the probability of default for a particular borrower at a given point in time, given input variables from their credit reports.

We start by formalizing our prediction problem. We adopt a discrete-time formulation for periods 0,1,...,T, each corresponding to a quarter. We let the variable D_t^i prescribe the state at time t for individual i with $\mathrm{D} \subset \mathbb{N}$ denoting the set of states. We define $D_1^i = 1$ if a consumer is 90+ days past due on any trade and $D_1^i = 0$ otherwise. Consumers will transition between these two states over their lifetime.

Our target outcome is 90+ days past due in the subsequent 8 quarters, defined as:

$$Y_t^i = \begin{cases} 0 & \text{if } \sum_{n=t}^{t+7} D_n^i = 0 \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

We allow the dynamics of the state process to be influenced by a vector of explanatory variables $X_{t-1}^i \in \mathbb{R}^{dx}$, which includes the state D_{t-1}^i. In our empirical implementation, X_{t-1}^i represents the features in Table 9. We fix a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and an information filtration $(\mathcal{F}_t)_{(t=0,1,\dots,T)}$. Then, we specify a probability transition function $h_\theta : \mathbb{R}^{dx} \to [0,1]$ satisfying

$$\mathbb{P}[Y_t^i = y | \mathcal{F}_{t-1}] = h_\theta(X_{t-1}^i), y \in D \tag{2}$$

where θ is a parameter to be estimated. Equation 2 gives the marginal conditional probability for the transition of individual i's debt from its state D_{t-1}^i at time $t-1$ to state y at time t

given the explanatory variables X^i_{t-1}.[8] Let g denote the standard softmax function:

$$g(z) = \left(\frac{1}{1+e^{-z}}\right), z \in \mathbb{R}^K, \tag{3}$$

where $K = |D|$. The vector output of the function g is a probability distribution on D.

The marginal probability defined in equation 2 is the theoretical counterpart of the empirical transition matrix reported in Table 2. We propose to model the transition function h_θ with a hybrid deep neural network/gradient boosting model, which combines the predictions of a deep neural network and an extreme gradient boosting model. We explain each of the component models and their properties and the rationale for combining them below.

4.1 Deep Neural Network

One component of our model is based on deep learning, in the class used by Sirignano, Sadhwani, and Giesecke (2018). We restrict attention to feed-forward neural networks, composed of an input layer, which corresponds to the data, one or more interacting hidden layers that non-linearly transform the data, and an output layer that aggregates the hidden layers into a prediction. Layers of the networks consist of neurons with each layer connected by synapses that transmit signals among neurons of subsequent layers. A neural network is in essence a sequence of nonlinear relationships. Each layer in the network takes the output from the previous layer and applies a linear transformation followed by an element-wise non-linear transformation.

4.2 eXtreme Gradient Boosting (XGBoost)

The second component of our model is Extreme Gradient Boosting, which builds on decision tree models. Tree-based models split the data several times based on certain cutoff values in the explanatory variables.[9] Gradient Boosted Trees (GBT) are an ensemble learning method that corrects for tree-based models' tendency to overfit to training data by recursively combining the forecasts of many over-simplified trees. Though shallow trees are "weak learners" on their own with little predictive power, the theory behind boosting proposes that a collection of weak learners, as an ensemble, creates a single strong learner with improved stability over a single complex tree. For a more detailed description of our model components see Appendix B.

[8] The state y encompasses realizations of the state between time t and $t+7$.

[9] Splitting means that different subsets of the dataset are created, where each observation belongs to one subset.

4.3 Hybrid DNN-GBT Model

We examined two techniques to create a hybrid DNN-GBT ensemble model. Ensemble models combine multiple learning algorithms to generate superior predictive performance than could be obtained from any of the constituent learning algorithms alone. The first method combines the two models by replacing the final layer of the neural network with a gradient boosted trees model. Examples of this approach are Chen, Lundberg, and Lee (2018) and Ren et al. (2017). The second, uses both models separately and then averages out the final predicted probabilities of the two models. We found the latter to perform better on our dataset. This method is similar to Kvamme et al. (2018), who combined a convolutional neural network with a random forest by averaging. Thus, our methodology relies on combining the output of the deep neural network with the output of a gradient boosted trees model. This is achieved in two steps:

1. For each observation, run DNN and GBT separately and obtain predicted probabilities for each of the models;
2. Take a weighted average of the predicted probabilities.[10]

4.4 Implementation

Table 9 lists the features from the credit report data we use as inputs in the model. They include information on balances and credit limits for different types of consumer debt, severity and number of delinquencies, credit utilization by type of product, public record items such as bankruptcy filings, collection items, and length of the credit history. In order to be consistent with the restrictions of the Fair Credit Reporting Act on 1970 and the Equal Opportunity in Credit Access Act of 1984 we do not include information on age or zip code, and we do not include any information on income, to be consistent with current credit scoring models. Table 9 lists the full set of features used in our machine learning models. We describe the rationale behind our feature selection in Appendix C. Section A.3 provides a comprehensive performance assessment of our model, Section A.4 uses a variety of interpretability techniques to understand which factors are strongly associated with default behavior, while Appendix D compares it to other approaches.

[10] We have investigated several weighting schemes, and the results are reported in Table 21.

5 Comparison with Credit Score

In this section, we compare the performance of our hybrid model to a conventional credit score.[11] The credit score is a summary indicator intended to predict the risk of default by the borrower and it is widely used by the financial industry. For most unsecured debt, lenders typically verify a perspective borrower's credit score at the time of application and sometimes a short recent sample of their credit history. For larger unsecured debts, lenders also typically require some form of income verification, as they do for secured debts, such as mortgages and auto loans. Still, the credit score is often a key determinant of crucial terms of the borrowing contract, such as the interest rate, the downpayment or the credit limit. We have access to a widely used conventional credit score that uses information from the three credit bureaus.

5.1 Ranking

A common way to measure the accuracy of conventional credit scoring models is the Gini coefficient, which measures the dispersion of the credit score distribution and therefore its ability to separate borrowers by their default risk. The Gini coefficient is related to a key performance metric for machine learning algorithm, the AUC score, with $Gini = 2 * AUC - 1$, so we can compare the performance of the credit score to our model along this dimension. Figure 11 plots the Gini coefficient for the credit score and our predicted default probability by quarter. The Gini coefficient for our model is about 0.85 between 2006Q1 and 2008Q3, and then rises to 0.86. For the credit score, the Gini coefficient is close to 0.81 until 2012Q3 when it drops to approximately 0.79 until the end of the sample, suggesting a drop in performance of the credit score in the aftermath of the Great Recession.

Table 3 shows the relationship between credit score, predicted probability and realized default rate, where default is defined as usual as 90+ days delinquency in the subsequent 8 quarters. The calculation proceeds as follows. We first compute the number of unique credit scores in the data. We create the same number of bins of equal size in our predicted probability distribution, and calculate the realized frequency of 90+ days delinquencies in the subsequent 8 quarters for each of these bins. Since higher credit scores correspond to lower probability of default, we present the negative of the rank correlation with realized defaults for the credit score. The results indicate that even though credit score is successful in rank-ordering customers by their future default rates, with rank correlations between 0.980 and 0.994, our deep neural network performs better, with rank correlations always at

[11] The hybrid model forecasts for each quarter are obtained using out-of-sample input data, as reported in Table 11.

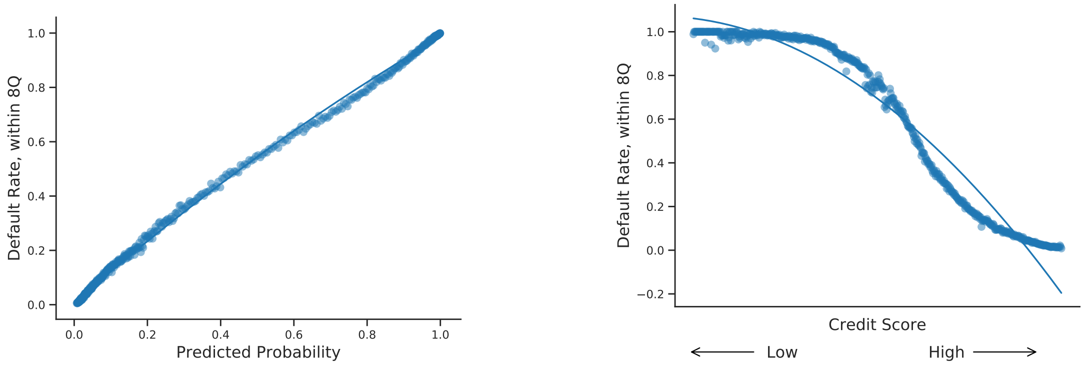
0.999. Figure 11 in Section A.5 plots the time series of these rank correlations by quarter for the entire sample period. The figure shows that the rank correlation for the predicted probability of default generated by our model is remarkably stable over time, while for the credit score it fluctuates from lows of around 0.975 before 2012 to a peak go 0.995 in 2013Q2 with notable quarter by quarter variation. This property of credit scores may be due to the fact that the credit score is an ordinal ranking and its distribution is designed to be stable over time, even if default risk at an individual or aggregate level may change substantially. Figure 10 in Section A.5 displays the histogram of credit score distributions in our sample for selected years, and show that these distributions are virtually identical over time.

Table 3: Borrower Rankings

Metric	2006	2007	2008	2009	2010	2011	2012	2013
PANEL A: Full Sample								
Rank Correlation								
Credit Score	0.9881	0.9804	0.9882	0.9816	0.9825	0.9861	0.9906	0.9944
Predicted Probability	0.9992	0.9994	0.9994	0.9993	0.9993	0.9993	0.9992	0.9992
GINI Coefficient								
Credit Score	0.8108	0.8142	0.8137	0.8143	0.8078	0.8008	0.7942	0.7898
Predicted Probability	0.8530	0.8529	0.8527	0.8598	0.8606	0.8592	0.8579	0.8563
PANEL B: Current Borrowers								
Rank Correlation								
Credit Score	0.9670	0.9613	0.9445	0.9653	0.9683	0.9585	0.9806	0.9559
Predicted Probability	0.9977	0.9983	0.9984	0.9978	0.9977	0.9977	0.9969	0.9973
GINI Coefficient								
Credit Score	0.6933	0.6935	0.6908	0.6807	0.6777	0.6833	0.6795	0.6810
Predicted Probability	0.7357	0.7242	0.7207	0.7178	0.7230	0.7324	0.7342	0.7373

Notes: Rank correlation between credit score, predicted probability of default according to our model and subsequent realized default frequency by year. Panel B only includes current borrowers, i.e., borrowers with no delinquencies. For the credit score, report the rank correlation between each unique value of the score and the default frequency. For predicted probability of default based on our hybrid DNN-GBT model, we first generate a number of bins equal to the number of unique credit score realizations in the data and then calculate the realized default frequency for each bin. Source: Authors' calculations based on Experian Data.

Panel B of Table 3 reports the rank correlation with the realized default rate and the Gini coefficients by year for the credit score and the probability of default predicted by our model restricting attention to the current population, that is those borrowers who do not have any outstanding delinquencies in the quarter of interest. The rank correlation between the credit score and the realized default rate drops by 1-3 percentage points for these borrowers, whereas for our model it drops by less than a quarter of 1 percent. The Gini coefficient drops from 80-81% to 68-69% for the credit score and from 85-86% to 72-74% for our predicted probability. These results suggest that when measured on the population of current borrowers, the performance advantage of our model relative to a conventional credit



(a) Predicted Probability of Default

(b) Credit Score

Figure 3: Realized Default Rates and Model Predicted Default Probability: Scatter Plot

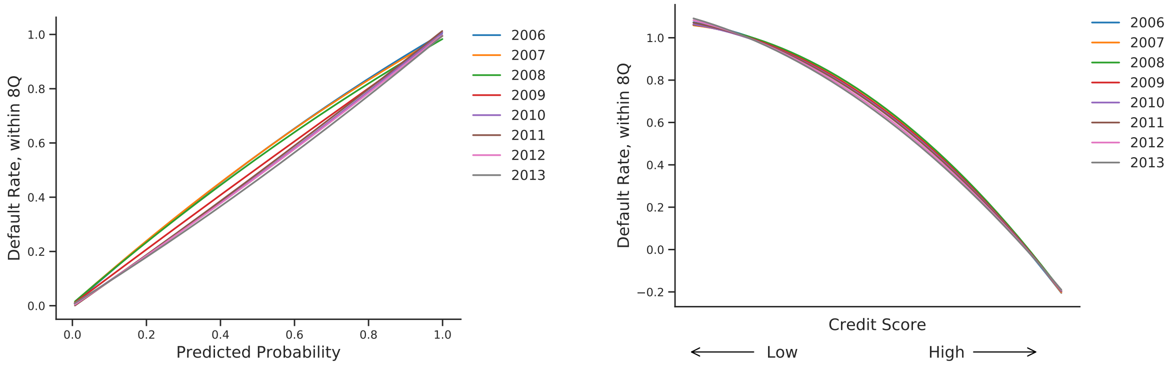
Notes: Scatter plot of realized default rates against model predicted default probability (a) and the credit score (b), with associated second-order polynomial fitted approximations for the year 2008. Source: Authors' calculations based on Experian Data.

score grows.[12]

Figure 3 plots a scatterplot of the realized default rate against the credit score (left panel) and our predicted probability (right panel) for all quarters in the year 2008. In addition to the raw data, we also plot second-order polynomial-fitted curves to approximate the relationship. The scatter plots of realized default rates against the predictions from our hybrid model lay mostly on the 45 degree line, consistent with the very high rank correlations reported in Table 3. By contrast, the relation between realized default rates and credit scores has an inverted S-shape, with the realized default rate equal to one for a large range of low credit scores and equal to zero for a large range of low credit scores, and a large variation only for intermediate credit scores.

Figure 4 plots second-order polynomial-fitted curves approximating the relation between realized default rates and those predicted by our model and the credit score for all years in which our model prediction is available, starting in 2006 until 2013, to examine how the relation between realized and predicted defaults varies with aggregate economic conditions. For the years at the height of the Great Recession, the default rate seems to be somewhat higher than our model prediction, but in all years the relation is very close to a 45 degree line. By contrast, there is virtually no change in the relation between the realized default rate and the credit score. This is by construction, since the distribution of credit scores is

[12] Appendix A.5 also plots the time series of the rank correlation and the Gini coefficient for the credit score and our model for the current population. The credit score shows a large drop in these statistics for the credit score during the Great Recession whereas for our model they are both stable over time. This is consistent with the notion that the performance of the credit score dropped during the 2007-2009 period.



(a) Predicted Probability of Default

(b) Credit Score

Figure 4: Realized Default Rates and Model Predicted Default Probability: Polynomial Approximation

Notes: Second-order polynomial approximation of the relationship between realized default rates against model predicted default probability (a) and the credit score (b) for selected years. Source: Authors' calculations based on Experian Data.

designed to only provide a relative ranking of default risk across borrowers.[13] This property of the credit score implies that it is unable to forecast variations in aggregate default risk. In Section 6.1, we will show that our model is able to capture variations in aggregate default risk while retaining a consistent ability to separate borrowers by their individual default risk.

We next examine how the ranking of borrowers varies under credit score and our model to understand the differences in performance under the two approaches. To do so, we consider the industry classification of borrowers into five risk categories Deep Subprime, Subprime, Near Prime, Prime and Super Prime.[14] As shown in Table 4, these categories account for respectively, 6.5%, 21.2%, 14.1%, 33.3% and 24.9% of all borrowers. We then create 5 correspondingly sized bins in our predicted probability of default for each quarter separately with bin 1 corresponding to the 6.5% of borrowers with the highest predicted default risk and bin 5 to the 24.9% of all borrowers with the lowest predicted default risk. Finally, we calculate the fraction of borrowers in each credit score category that is in each of the 5 predicted default risk categories and their realized and predicted default rate. The results are displayed in Table 4. We also report the average realized and predicted default rate for each credit score category overall (columns 7 and 8) and for each predicted default risk category for all credit score (last 5 rows).

[13] Credit scores are specifically designed to provide a stable ranking by using multiple years of data.

[14] The threshold levels for these categories are: 1) Deep Subprime: up to 499 credit score; 2) Subprime: 500-600 credit score; 3) Near Prime: 601-660 credit score; 4) Prime: 661-780 credit score; 5) Super Prime: higher than 781 credit score.

Table 4: Credit Risk Differences

Credit Score		Predicted Default Probability		Default Rate		Average Default Rate	
	Share		Share	Realized	Predicted	Realized	Predicted
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Deep Subprime	6.5%	1	49.5%	99.46%	99.20%	95.45%	94.40%
		2	49.5%	92.10%	90.47%		
		3	0.9%	63.69%	52.71%		
		4	0.0%	41.98%	13.79%		
		5	0.0%	37.14%	2.11%		
Subprime	21.2%	1	14.5%	99.21%	98.97%	78.64%	77.39%
		2	64.9%	84.34%	84.04%		
		3	16.7%	52.01%	46.78%		
		4	3.9%	21.58%	17.88%		
		5	0.0%	17.28%	2.46%		
Near Prime	14.1%	1	1.2%	98.97%	98.80%	43.71%	42.79%
		2	24.3%	76.60%	78.78%		
		3	43.0%	41.44%	40.59%		
		4	30.8%	19.69%	16.20%		
		5	0.7%	3.37%	2.93%		
Prime	33.3%	1	0.1%	98.82%	98.80%	14.31%	14.59%
		2	2.4%	75.15%	77.67%		
		3	13.1%	33.55%	36.68%		
		4	71.3%	10.67%	10.51%		
		5	13.1%	3.21%	2.67%		
Super Prime	24.9%	1	0.0%	99.04%	98.95%	2.56%	2.80%
		2	0.1%	81.19%	78.17%		
		3	0.2%	32.03%	34.54%		
		4	17.6%	5.56%	6.14%		
		5	82.1%	1.75%	1.92%		
All		1	6.5%	99.33%	99.08%		
		2	21.2%	83.92%	83.92%		
		3	14.1%	41.70%	40.96%		
		4	33.3%	11.45%	10.86%		
		5	24.9%	2.01%	2.05%		

Notes: Borrowers are classified into 5 categories of default risk standard in the industry named in column (1). The threshold levels for these categories are: 1) Deep Subprime: up to 499 credit score; 2) Subprime: 500-600 credit score; 3) Near Prime: 601-660 credit score; 4) Prime: 661-780 credit score; 5) Super Prime: higher than 781 credit score. The fraction of borrowers in each category is reported in column (2). Borrowers are also assigned to 5 categories of predicted default risk based on our hybrid model from the highest default risk (1) to the lowest (5), where the share of borrowers in each predicted default risk category is the same as for the credit score categories Deep Subprime to Super Prime. For each credit score risk category the share of borrowers in each predicted default risk category is reported in column (4). Columns (5) and (6) report the corresponding realized and predicted default probability for each credit score category interacted with predicted default risk category. Columns (7) and (8) report the average realized and predicted default probability for each credit score category. All rates, fractions and shares in percentage. Total # of observations: 17,732,772. Time period 2006Q1-2013Q4. Source: Authors' calculations based on Experian Data.

These results suggest that our model does well in predicting the default probability of borrowers in all categories, with a slight tendency to under-predict the probability of default by 1-5 percentage points for the Deep Subprime and Subprime borrowers. The majority of Deep Subprime borrowers fall in the two lowest categories of predicted default risk. For Subprime borrowers, 65% fall into the corresponding second category of default risk, while 15% fall in the first and 17% into the third. This corresponds to a sizable discrepancy as the average realized default probability for Subprime borrowers is 79%, whereas it is 95% for those in the first category and only 44% for those in the third. By contrast, the predicted default risk is very close to the realized default risk for Subprime borrowers in all categories of the predicted default risk distribution, with a discrepancy under 1 percentage point for predicted default risk category 1 and 2, and around 5 percentage points for category 3 and 4. Near Prime borrowers also display a wide dispersion across predicted default risk categories with only 43% falling into the corresponding third category, 24% falling in category 2 (higher default risk) and 31% falling into category 4 (lower default risk). Again, the realized default rates vary substantially for Near Prime borrowers by predicted default risk category, from 77% in category 2, to 41% and 20% in category 3 and 4, respectively, while the predicted default risk in much closer to the realized, with a maximum 3 percentage point discrepancy.

The discrepancy in classification for the credit score are lower for Prime and Super Prime borrowers. 13% of Prime borrowers fall into category 3 (higher default risk), 13% in category 5 (lower default risk) and 71% in the corresponding category 4. The realized default rates are 11% for Prime borrowers in category 4, and 34% and 3% respectively for Prime borrowers in category 3 and 5. Only 18% of Super Prime borrowers fall in category 4 of predicted default risk (higher risk) and 82% fall in the corresponding category 5. Moreover, the differences in realized default risk between these categories are minor, with a realized default rate of 6% and 2% for categories 4 and 5, respectively. These results suggest that credit scores misclassify borrowers across risk categories with very different realized default rates. By contrast, as shown in the bottom 5 rows of Table 4 and by columns (6) and (8), our model is very successful at predicting the default rate for borrowers irrespective of their credit score.

5.2 Feature Attribution

We next investigate feature attribution differences across credit scores and our hybrid model. We grouped our features into five categories to correspond to information we obtained from marketing resources for the credit score, and aggregated the absolute value of the SHAP values for each instance across each categories across our testing dataset for the pooled model. These categories are, payment history, amount owed, length of credit history, credit mix and new credit. Their contribution towards credit scores is reported in Table 5.

Table 5: Feature Attribution Differences

		Model	
Feature Group	# of Features	Hybrid	Credit Score
Payment History	23	0.32	0.35
Accounts Owed	50	0.50	0.3
Length of Credit	6	0.09	0.15
Debt Products	5	0.06	0.1
Inquiries	4	0.03	0.1

Notes: This table reports the Shapley values for five feature groups across four models. For each prediction window, we compute the Shapley value for each of the observations and for each feature. We then calculate the sum of the absolute value for each feature, aggregate it across the feature groups and report the results for the group. We normalized the results so that for each model the four groups sum up to 1. Source: Authors' calculations based on Experian data.

Contrary to credit scores, features relating to credit inquiries, debt products, and length of credit history account for 18% of the total variation in predicted probabilities for our hybrid model. The aggregate impact of these three factors is approximately half of the variation they explain of credit scores, which can partly be attributed to the low number of features we include in our models pertaining to these three groups. However, notice that even the per feature contribution is low for inquiries for our hybrid model.[15] Next, payment history accounts for 32% of the variation in predicted probabilities, being 3% short of its contribution towards credit scores. Perhaps most strikingly, accounts owed explain 50% of the variation for our hybrid model, while only 30% for credit scores. This exercise once again illustrates that features relating to debt balances are the most important determinants for our model's output, contrasted with credit scores, where payment history is registered as the most important predictor.

5.3 Vulnerable Populations

There has been a growing concern about the differential impacts of improved statistical technologies across categories such as age, race, gender, and income group. For instance, Fuster et al. (2018) found that though each group gains from improved predictive accuracy in creditworthiness (i.e., probability of mortgage default), Black and White Hispanic borrowers are predicted to lose, relative to White and Asian borrowers. They identified increased flexibility as the reason behind the unequal distribution of gains. Motivated by this finding,

[15] The per feature contribution for credit inquiries is 0.006, contrasted with 0.01, 0.013, 0.014 and 0.016 for amounts owed, debt products, payment history, and length of credit respectively.

we compare the impacts of our technology relative to credit scores across age and income categories. To do so, we first rank our borrowers based on their percentile position in the predicted probability and credit score distribution respectively, and compute the difference.[16]. Then, to investigate which model provides better credit market access to the most vulnerable subgroup, the young with low income, we run regressions of the form:

$$Y_{ist} = \alpha + \beta_1 x_{1,ist} + \beta_2 x_{2,ist} + \beta_3 x_{1,ist} \times x_{2,ist} + \lambda_t + \delta_s + \delta_{st} + \epsilon_{ist} \tag{4}$$

where λ controls for any time varying changes common to all individuals (e.g., such as the Credit Card Act of 2009), while δ controls for any time invariant differences and for any arbitrary trends across states (e.g., differential evolution of default rates across states over time). We include indicator variables for being under 30 years old (Young), for being in the bottom quintile of the income distribution (Income$_{p20}$), and for being in a county with above the median percentile unemployment shock in the quarter (e.g., measured by the difference between the unemployment rate in county c in quarter q and the average unemployment rate between 2000 and 2005 in county c). In most our specifications, we control for defaulting in the subsequent two years (Default) to ensure that differences are driven primarily by giving better access to credit to those who do not default on their loans.

We report descriptive statistics for this exercise in Table 6. We can see that young borrowers default on their loans at slightly higher rates, have lower debt and 90+ dpd debt balances, lower household income, and ranked lower in both the credit scores and our model's risk distribution. The cross-group differences are similar in direction but even larger in magnitude for the first income quantile vs. the rest comparison.

To complement Table 6, Panel (a) of Figure 5 plots the average difference across age and across default status, while Panel (b) displays these differences over time. We can see that the introduction of our hybrid model would mostly benefit young borrowers. For instance, the raw difference of 2 units for the youngest age group translates into improved credit ranking by 2 percentiles, which would improve their access to credit markets. Additionally, individuals who do not default would on average benefit across all age groups.

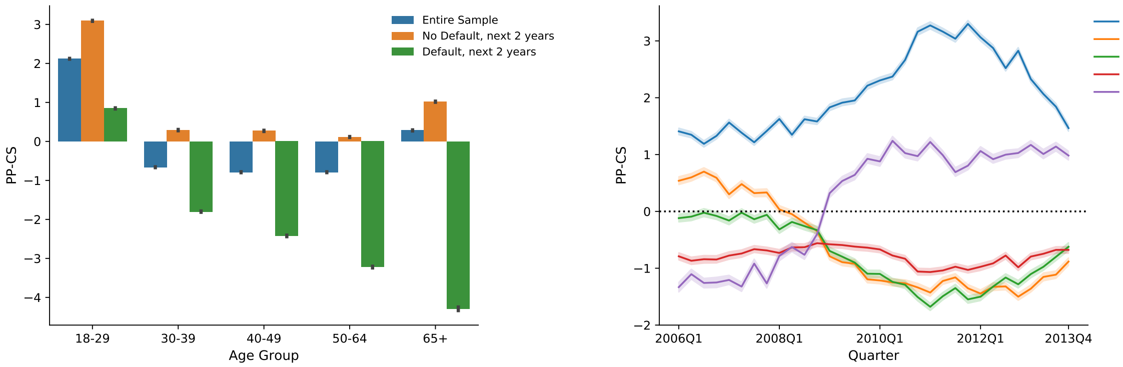
We report the regression results in Table 7. Columns (1) shows that precisely the most vulnerable subgroup, young individuals whose income is in the bottom quintile of the income distribution benefit the most from our credit risk model. In particular, our model would rank these individuals by 3.3 percentile higher on average in the credit risk distribution. We then control for default status, and while Column (2) shows that every borrower who do not end up defaulting would gain from our credit risk assessment, young and low income no-defaulters

[16] Note that positive values imply lower credit risk by our model.

Table 6: Descriptive Statistics (Vulnerable Populations)

	Age: < 30	Age ≥ 30	t-test$_{Age}$	Income$_{p20}$	Income$_{>p20}$	t-test$_{Income}$
Default	0.43	0.32	-0.11***	0.65	0.27	-0.39***
	(0.50)	(0.47)		(0.48)	(0.44)	
Predicted Probability	39.26	53.49	14.23***	25.93	56.76	30.83***
	(22.76)	(29.57)		(18.49)	(27.75)	
Credit Score	37.14	54.05	16.91***	23.79	57.37	33.58***
	(22.94)	(29.24)		(17.79)	(27.19)	
Total debt ($)	25.38	93.15	67.77***	10.57	96.76	86.18***
	(68.87)	(192.75)		(40.29)	(192.98)	
90+ dpd debt ($)	1.51	4.11	2.60***	2.17	3.94	1.77***
	(16.23)	(39.93)		(16.27)	(39.88)	
Household Income ($)	61.11	99.33	38.22***	29.06	107.21	78.14***
	(423.20)	(315.58)		(6.43)	(380.03)	
Age	24.38	51.74	27.36***	32.22	49.62	17.40***
	(3.25)	(13.44)		(12.54)	(15.31)	
Unemployment Shock	2.15	2.21	0.06***	2.18	2.21	0.03***
	(2.37)	(2.39)		(2.43)	(2.37)	
N	3723837	14008935	17732772	3557963	14066498	17624461

Notes: *** denotes statistical significance at the 1% level. Income and debt variables in thousands.



(a) Pooled

(b) Time Series

Figure 5: Differences in Creditworthiness by Age across and over Time

Notes: Source: Authors' calculations based on Experian Data.

Table 7: Vulnerable Populations

	(1)	(2)	(3)	(4)
Dependent Variable: Predicted Probability$_t$ - Credit Score$_t$				
Young	0.9633***	1.0077***	1.0273***	0.6923***
	(0.0350)	(0.0334)	(0.0449)	(0.0643)
Income$_{p20}$	1.5605***	3.2570***	3.2147***	2.5839***
	(0.0432)	(0.0473)	(0.0482)	(0.0624)
Young × Income$_{p20}$	1.4928***	0.7752***	0.9024***	1.0064***
	(0.0380)	(0.0346)	(0.0374)	(0.0526)
Default		-3.4441***	-3.4437***	-2.7325***
		(0.0440)	(0.0439)	(0.0601)
Unemployment Shock			0.1879***	0.4456***
			(0.0298)	(0.0448)
Young × Unemployment Shock			-0.0328	-0.1474**
			(0.0507)	(0.0730)
Income$_{p20}$ × Unemployment Shock			0.0955	-0.2791***
			(0.0611)	(0.0762)
Young × Income$_{p20}$ × Unemployment Shock			-0.2564***	-0.0522
			(0.0513)	(0.0679)
Constant	-0.7399***	0.1782***	0.0842***	-0.0560*
	(0.0122)	(0.0116)	(0.0186)	(0.0297)
State Fixed Effects	X	X	X	X
Quarter Fixed Effects	X	X	X	X
State X Quarter Fixed Effects	X	X	X	X
Sample: 2007-2009				X
Observations	17624453	17624453	17395749	6327506
R^2	0.0191	0.0373	0.0374	0.0276

Notes: This table reports regressions of the form specified by Equation (4). Model (1) does not control for default, Model (2) adds an indicator variable for default, while Model (3) and (4) include an indicator variable of being in a county with over the median unemployment shock as an additional dimension of interaction. We measure unemployment shock by the difference between unemployment rate in county c in quarter t and the average unemployment rate in county c between 2000 and 2005. Standard errors are clustered by state and quarter. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$. To mitigate the influence of outliers, we winsorized are dependent variable at the at the 0.1 and 99.9 percentiles.

would be the largest beneficiaries, being ranked 5.2 percentile higher by our hybrid model. Columns (3-4) show that the benefits are similar in areas hard hit by unemployment shocks. This result holds during the Great Recession (i.e, 2007-2009), and the interpretation of this is that, relative to other groups, our model provides more favorable credit risk assessment for young and low income individuals, even in areas experiencing severe financial distress. Since credit scores do not provide an exact probability of default, we are unable to conclude whether our model would provide a larger number of borrowers access to credit (i.e., the extensive margin) across age and income groups in the borrower population. However, contrasting the relative ranking reiterates the notion that credit scores are indiscriminately low for young and low income borrowers.

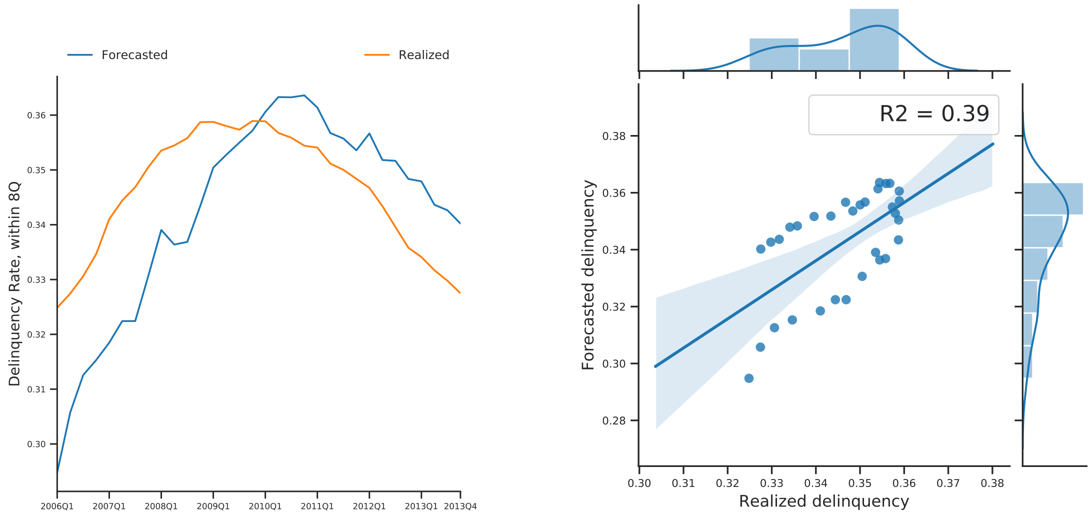
6 Applications

In this section, we use our model in two applications. We first show that our model is able to accurately predict variations in aggregate default risk, and second, we illustrate the value added for lenders and borrowers from our hybrid model.

6.1 Predicting Systemic Risk

We first analyze the aggregate forecasting power of our hybrid model. We aggregate the deep-learning forecasts for individual accounts to generate macroeconomic forecasts of credit risk by taking the average of the predicted probabilities over a given forecast period. Since our sample of consumers in nationally representative in each quarter, this will provide an unbiased estimate of the aggregate default risk predicted by our model. We calculate the aggregate default probability for 2006Q1-2013Q4, and show that our model is able to predict the spike in delinquencies during the 2007-2009 financial crisis and also the reduction in delinquencies since then. This estimate of aggregate default risk could be used as a proxy of systemic risk in the household sector. The results are displayed in Figure 6.

Panel (a) plots the aggregate predicted default rate from our hybrid model and compares it to the aggregate realized default rate. While our predicted aggregate default rate is approximately 2 percentage points lower than the realized in 2006 and 2007, it rises at a similar speed as the realized default rate. It peaks in 2010Q2, approximately 2 quarters after the peak in the realized rate and then declines in the ensuing period, again reflecting the behavior of the realized rate, though it overestimates it by about 1 percentage point. Panel (b) shows a scatter plot of the predicted aggregate default rate against the realized for the different quarters in our sample period. The correlation between the predicted and realized



(a) Predicted and realized

(b) Correlation

Figure 6: Consumers with 90+ Days Delinquency: Predicted vs. Realized

Notes: Consumers with 90+ Days Delinquency within the Subsequent 8 Quarters. Aggregate default rates are obtained by averaging across all consumers in each period. Source: Authors' calculations based on Experian Data.

aggregate default rate is 62%.

6.2 Value Added

We assess the economic salience of our hybrid DNN-GBT model by analyzing its value added for lenders and borrowers. For lenders, we examine the role our model can play in minimizing the losses from default. For borrowers, we calculate the interest savings for borrowers who are misclassified as having an excessively high probability of default based on the credit score compared to our model.

6.2.1 Lenders

We follow the framework proposed by Khandani, Kim, and Lo (2010), which compares the value of having a prediction of default risk to having none, and we make the same simplifying assumptions with respect to the revenues and costs of the consumer lending business. Specifically, in absence of any forecasts, it is assumed a lender will take no action regarding credit risk, implying that customers who default will generate losses for the lender, and customers who are current on their payments will generate positive revenues from financing fees on their running balances. To simplify, we assume that all defaulting and non-defaulting customers have the same running balance, B_r, but defaulting customers increase their balance to B_d prior to default. We refer to the ratio between B_d and B_r as "run-up." It is assumed that with a model to predict default risk, a lender can avoid losses of defaulting customers by cutting their credit line and avoiding run-up. Then, the value added as proposed by Khandani, Kim, and Lo (2010) can be written as follows:

$$VA(r, N, TN, FN, FP) = \frac{TN - FN\big[1 - (1+r)^{-N}\big]\big[\frac{B_d}{B_r} - 1\big]^{-1}}{TN + FP} \tag{5}$$

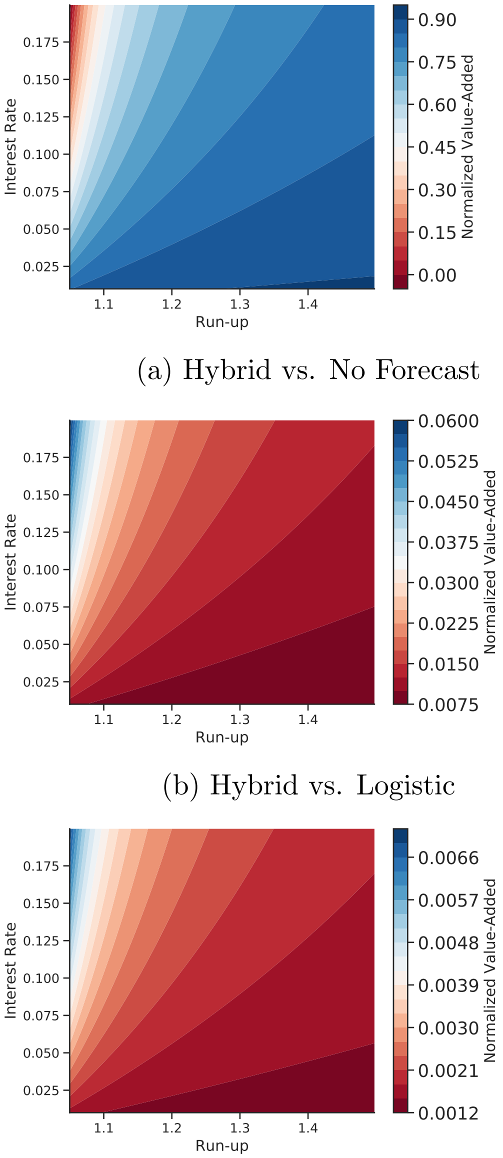
where r refers to the interest rate, N the loan's amortization period, and TN, FN, FP refer to true negatives, false negatives and false positives respectively. Panel (a) of Figure 7 plots the Value Added (VA) as a function of interest rate and the ratio of run-up balance for our out-of-sample forecasts of 90+ days delinquencies over the subsequent 8 quarters for 2012Q4. These estimates imply cost savings of over 60% of total losses when compared to having no forecast model for a run-up of 1.2 at a 10% interest rate for an amortization period for 3 years.

We next compare the value added of our hybrid model with default predictions generated by a logistic regression. This exercise illustrates the gains from adopting a better technology for credit allocation. Panel (b) of Figure 7 shows more modest, but substantial cost savings in the range of 1-6% and approximately 2.5% for a 1.2 run-up at a 10% interest rate with a 3 year amortization period. Panel (c) calculates the cost savings associated to using our hybrid model in comparison to random forest. In this case the cost savings range from 0.1-0.7%. This exercise then confirms the advantages of using deep learning over other technologies in predicting default.

6.2.2 Borrowers

We now examine the potential cost savings for consumers who would be offered credit according to the predicted default probability implied by our model instead of a conventional credit score. Following our approach in Section 5, we create credit score categories based on common industry standards and corresponding predicted probability bins with the same number of observations for each quarter, and we place customers in these bins. The distribution of customers is summarized in Table 17. We then follow the information on interest rates by credit score category in Table 2 in Agarwal et al. (2015) to obtain the cost of credit on credit card balances. [17]

[17] Credit card interest rates are notoriously invariant to overall changes in interest rates, so the calculations reported in this section apply irrespective of the time period. See Ausubel (1991) and Calem and Mester (1995).



(a) Hybrid vs. No Forecast

(b) Hybrid vs. Logistic

(c) Hybrid vs. RF

Figure 7: Value-added of machine-learning forecasts

Notes: Value-added of machine-learning forecasts of 90+ days delinquency over 8Q forecast horizons on data from 2012Q4. VA values are calculated with amortization period N = 3 years and a 50% classification threshold. Source: Authors' calculations based on Experian Data.

To obtain the cost savings for consumers, we use the difference in interest rates by credit score category based on how they would be classified according to our model.[18] For customers who are placed in higher risk categories by the credit score compared to our predicted probability of default, interest rates on credit cards are higher than they would have been if they had been classified according to our model. Thus, using our model to score consumers rather than the credit score would generate the cost savings for them. For customers placed in risk categories by the credit score that are too low relative to the default risk predicted by our model, interest rates will be higher under our model. The calculation is made for each individual consumer. The average for each credit score category is then computed. The information on interest rates and balances, and the dollar value of cost savings for different credit card categories is reported in Table 8. We report this in current USD terms, since annual interest rate savings are symmetric by definition. The largest gains accrue to customers with Subprime and Near Prime credit scores. As we showed in Section 5, they are more likely to be attributed a probability of default by the credit score that is too low compared to our model predictions. Additionally, the biggest variation in credit card interest rates occurs across Subprime and Near Prime borrowers in comparison to Prime based on Agarwal et al. (2015). The cost savings for these borrowers average out to $1,149-1,403. Gains for Prime and Super Prime borrowers who are attributed a lower default probability by our model are very modest, as credit card interest rates vary little by credit score for Prime and Superprime borrowers. On the other hand, Prime and Superprime borrowers who are placed in group 1, corresponding to the highest predicted default probability based on our model face, substantial losses in the order of 4-5% of total credit card balances or $268-411. The cumulated interest rate cost savings across all consumers in our sample is $762,205,377, which amounts to $43 per capita.

This calculation provide us with a lower bound for the cost savings of being classified according to our model in comparison to the credit score, as they do not take into account the higher credit limits and potential behavioral responses of customers faced with higher borrowing capacity and lower interest rates. As shown in Agarwal et al. (2015), changes in the cost of funds for lenders mainly translate into changes in credit limits and exclusively for higher credit score borrowers. Therefore, being placed in a higher risk category for consumers also inhibits their ability to benefit from expansionary monetary policy. Additionally, we do not take into account the fact that more expensive credit in the form of higher interest rate costs makes it more likely that the consumer will incur missed payments in response to temporary changes in income. Fees for missed payments constitute a substantial component

[18] We draw interest rates from a truncated normal distribution with mean and standard deviation as in Agarwal et al. (2015).

Table 8: Cost of Credit Risk Misclassification

		Credit Score			
		Subprime, Near Prime	Prime Low	Prime Mid	Prime High, Superprime
		Annual Average Cost Saving ($)			
Predicted Default bin	1	0	-411.3	-268.2	-301.9
	2	1149.4	0	84.1	15.5
	3	1403.1	-170.9	0	-54.5
	4	1273.6	-49.2	118.6	0
		Annual Average Cost Saving / Income (%)			
Predicted Default bin	1	0	-0.57	-0.34	-0.31
	2	1.44	0	0.11	0.02
	3	1.44	-0.19	0	-0.06
	4	1.07	-0.04	0.11	0
		Annual Average Cost Saving / Debt (%)			
Predicted Default bin	1	0	-2.6	-2.6	-2.9
	2	3.8	0	0.5	0.1
	3	2.7	-0.6	0	-0.4
	4	2.7	-0.1	0.4	0

Notes: This table reports the average cost savings for consumers across credit score and predicted default probability bins for our sample. The cumulative savings for consumers on both credit card and bankcard debt adds up to $762,205,377. Time period 2006Q1-2013Q4. Source: Authors' calculations based on Experian Data.

of credit card costs for consumers, and the ability to avoid these fees would contribute to substantial cost savings for consumers (see Agarwal et al. (2014)).

7 Conclusion

We have proposed to use deep learning to develop a model to predict consumer default. Our model uses the same data used by conventional scoring models and abides with all legislative restrictions in the United States. We show that our model compares favorably to conventional credit scoring models in ranking individual consumers by their default risk, and is also able to capture variations in aggregate default risk. Our model is interpretable and allows to identify the factors that are most strongly associated with default. Whereas conventional credit scoring models emphasize utilization rates, our analysis suggests that the number and balances on open trades are the factors which associate more strongly to higher default probabilities. Our model is able to provide a default prediction for all consumers with a non-empty credit record. Additionally, we show that our hybrid DNN-GBT model performs better than standard machine learning models of default based on logistic regression and can accrue cost saving to lenders in the order of 1-6% compared to default predictions based on logistic regression, as well as interest rate cost savings for consumers of up to $1,401 per year.

A Appendix

A.1 Performance Metrics

Suppose a binary classifier is given and applied to a sample of N observations. For each instance i, let y_i denote the true outcome. For each observation, the model generates a probability that an observation with feature vector x_i belongs to class 1. This predicted probability, $f(x_i)$ is then evaluated based on a threshold to classify observations into class 1 or 0. Given a threshold level (c), let True Positive (TP) denote the number of observations that are correctly classified as type 0, True Negative (TN) be the number of observations that are correctly classified as type 1, False Positive (FP) be the number of observations that are type 1 but incorrectly classified as type 0, and, finally, False Negative (FN) be the number of observations that are actually of type 0 but incorrectly classified as type 0. Based on these definitions, one can define the following metrics to assess the performance of the classifier:

$$\text{True Negative Rate (TNR)} \equiv \frac{\text{TN}}{\text{TN+FP}} \tag{6}$$

$$\text{False Positive Rate (FPR)} \equiv \frac{\text{FP}}{\text{FP+TN}} \tag{7}$$

$$\text{Precision} \equiv \frac{\text{TP}}{\text{TP + FP}} \tag{8}$$

$$\text{Recall} \equiv \frac{\text{TP}}{\text{TP + FN}} \tag{9}$$

$$\text{F-measure} \equiv \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Precision + Recall}} \tag{10}$$

$$\text{Accuracy} \equiv \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{11}$$

$$\text{Youden's J statistic} \equiv \frac{\text{TP}}{\text{TP + FN}} + \frac{\text{TN}}{\text{TN + FP}} - 1 \tag{12}$$

$$\text{ROC AUC} = \int_{\infty}^{-\infty} \text{TPR}(c)\text{FPR}'(c)\text{dc} \tag{13}$$

$$\text{Cross-entropy loss} = -\frac{1}{N}\sum_{i=1}^{N}(y_i \cdot \log(f(x_i)) + (1 - y_i) \cdot \log(1 - f(x_i)) \tag{14}$$

A.2 Features

We summarize the list of features included in our model in Table 9, while Table 10 provides summary statistics for selected features.

Table 9: Model Inputs

Amount past due on bankcard trades presently 30 dpd
Amount past due on credit card trades presently 90+ dpd
Amount past due on installment trades presently 90+ dpd
Amount past due on joint mortgage type trades
Amount past due on revolving trades presently 30 dpd
Amount past due on revolving trades presently 90+ dpd
Amount past due on trades presently 30 dpd
Amount past due on trades presently 90+ dpd
Balance on authorized user trades
Balance on bankcard trades presently 90+ dpd
Balance on collections
Balance on collections, last 24 months
Balance on credit & bankcards
Balance on home equity line of credit trades
Balance on installment trades
Balance on installment trades presently 90+ dpd
Balance on joint installment trades
Balance on joint revolving trades
Balance on open auto loan trades
Balance on revolving trades presently 90+ dpd
Balance on second mortgage trades
Balance on trades presently 30 dpd
Balance on trades presently 60 dpd
Balance on trades presently 90+ days delinquent or derogatory
Bankcard inquiries made in the last 3 months
Credit amount on home equity line of credit trades
Credit amount on joint revolving trades
Credit amount on joint trades
Credit amount on open credit card trades
Credit amount on open deferred student trades
Credit amount on open non-deferred student trades
Credit amount on open trades
Credit amount on revolving trades
Credit amount paid down on open first mortgage trades
Credit card utilization
Fraction of 30 dpd debt
Fraction of 60 dpd debt
Fraction of 90+ days delinquent debt
Heloc utilization
Inquiries made in the last 12 months (no deduplication)
Installment utilization
Joint debt balances
Monthly payment on credit card trades
Monthly payment on debt
Monthly payment on joint installment trades
Monthly payment on joint mortgage type trades
Monthly payment on open first mortgage trades
Monthly payment on open non-deferred student trades
Monthly payment on second mortgage trades
Months since the most recent 30-180 days delinquency on auto loan or lease trades
Months since the most recent 30-180 days delinquency on credit card trades
Months since the most recent 30-180 days delinquency on trades
Months since the most recent 90+ days delinquency
Months since the most recent foreclosure proceeding started on first mortgage trades
Months since the most recently closed, transferred, or refinanced first mortgage trade
Months since the most recently opened credit card trade
Months since the most recently opened first mortgage trade
Months since the most recently opened home equity line of credit trade
Months since the oldest trade was opened
Mortgage to total debt
Mortgage type inquiries made inthe last 3 months
Number of auto loan trades
Number of collections
Number of credit & bankcards
Number of installment trades
Number of 90 days delinquencies in the last 36 months
Number of 90 days delinquencies in the last 6 months
Number of open mortgage type trades
Open home equity line of credit trades
Public record bankruptcies
Public record discharged bankruptcies
Public record dismissed bankruptcies
Public records filed in the last 24 months
Ratio of inquiries (no deduplication) to trades opened in the last 6 months
Total debt balances
Trades legally paid in full for less than the full balance
Unsatisfied collections
Utilization ratio
Worst ever status on a credit card trade in the last 24 months
Worst ever status on a mortgage type trade in the last 24 months
Worst ever status on a trade in the last 24 months
Worst ever status on an auto loan or lease trade in the last 24 months
Worst present status on a credit card trade
Worst present status on a mortgage type trade
Worst present status on a trade
Worst present status on a trade (excluding collections)
Worst present status on an installment trade
Worst present status on an open trade

Notes: List of features included in our model.

Table 10: Summary Statistics

Feature	Mean	Std. Dev	25%	Median	75%
Open home equity line of credit trades	0.11	0.33	0	0	0
Installment utilization	0.28	0.38	0	0	0.66
Mortgage to total debt	0.3	0.42	0	0	0.82
Credit card utilization	0.32	8.19	0	0.05	0.35
Number of auto loan trades	0.34	0.6	0	0	1
Number of open mortgage type trades	0.5	0.83	0	0	1
Unsatisfied collections	0.73	2.12	0	0	0
Number of installment trades	0.74	1.3	0	0	1
Inquiries made in the last 12 months (no deduplication)	1.38	2.28	0	1	2
Number of credit & bankcards	5.73	6.19	0	4	9
Months since the most recent 90+ days delinquency	16.16	18.32	3	9	24
Months since the most recent 30-180 days delinquency on trades	22.65	23.26	2	14	39
Months since the most recent 30-180 days delinquency on credit card trades	26.45	23.91	4	20	45
Months since the most recent 30-180 days delinquency on auto loan or lease trades	28.08	24.36	5	22	48
Months since the most recently closed, transferred, or refinanced first mortgage trade	39.51	30.85	14	32	60
Worst ever status on a credit card trade in the last 24 months	45.26	116.61	1	1	2
Months since the most recently opened home equity line of credit trade	65.19	48.74	28	57	90
Worst present status on a trade (excluding collections)	68.34	145.61	1	1	2
Months since the most recently opened first mortgage trade	70.86	69.13	23	51	95
Monthly payment on credit card trades	121.22	366.75	0	31	125
Worst ever status on a trade in the last 24 months	126.6	180.08	1	1	400
Months since the oldest trade was opened	196.45	126.35	98	178	271
Monthly payment on open first mortgage trades	474.67	2163.29	0	0	694
Balance on collections, placed with the collector in the last 24 months	560.26	3277.47	0	0	0
Monthly payment on debt	907.95	11059.39	20	333	1232
Credit amount on open non-deferred student trades	2123.32	11983.58	0	0	0
Balance on trades presently 90+ days delinquent or derogatory	3125.34	33186.26	0	0	0
Balance on joint installment trades	4142.13	26920.71	0	0	0
Balance on open auto loan trades	4472.88	11608.48	0	0	3915
Credit amount paid down on open first mortgage trades	6109.98	164527.99	0	0	2255
Balance on installment trades	8754.12	32554.07	0	0	10583
Balance on credit & bankcards	8808.63	19284.41	0	1526	8624
Credit amount on home equity line of credit trades	9139.8	47969.14	0	0	0
Credit amount on joint revolving trades	10698.74	44129.16	0	0	2200
Credit amount on open credit card trades	21475.9	30662.29	0	8600	31947
Credit amount on revolving trades	30517.5	62226.97	0	9500	37311
Joint debt balances	50957.9	143808.27	0	0	29546
Credit amount on joint trades	64191.25	220910.44	0	0	55100
Total debt balances	77126.36	170742.97	318	11738	95808
Credit amount on open trades	108480.31	259094	3000	33146	146535

A.3 Classifier Performance

In this section, we describe the performance of our hybrid model under various training and testing windows. First, we evaluate our model on the pooled sample (2004Q1-2013Q4), where we apply a random 60%-20%-20% split to our training, validation, and testing sets. Then, to account for look-ahead bias, we train and test our models based on 8 quarter windows that were observable at the time of forecast. In particular, we require our training and testing sets to be separated by 8 quarters to avoid overlap. For instance, the second out-of-sample model was calibrated using input data from 2004Q2, from which the parameter estimates were applied to the input data in 2006Q2 to generate forecasts of delinquencies over the 8 quarter window from 2006Q3-2008Q2. This gives us a total of 32+1 calibration and testing periods reported in Table 11. The percentage of 90+ days past due accounts within 8 quarters varies from 32.5% to 35.9%.

The hybrid model outputs a continuous variable that, under certain circumstances, can be interpreted as an estimate of the probability of an account becoming 90+ days delinquent during the subsequent 8 quarters. One measure of the model's success is its ability to differentiate between accounts that did become delinquent and those that did not; if these two groups have the same forecasts, the model provides no value. Table 11 presents the average forecast for accounts that did and did not fall into the 90+ days delinquency category over the 32+1 evaluation periods. For instance, during the testing period for 2010Q4, the model's average prediction among the 35.44% of accounts that became 90+ days delinquent was 73.12%, while the average prediction among the 64.56% of accounts that did not was 16.18%. We should highlight that these are truly out-of-sample predictions, since the model is calibrated using input data from 2008Q4. This shows the forecasting power of our model in distinguishing between accounts that will and will not become delinquent within 8 quarters. Furthermore, this forecasting power seems to be stable over the 32+1 calibration and evaluation periods, partly driven by the frequent re-calibration of the model that captures some of the changing dynamics of consumer behavior.

We also look at accounts that are current as of the forecast date but become 90+ days delinquent within the subsequent 8 quarters. In particular, we contrast the model's average prediction among individuals who were current on their accounts but became 90+ days delinquent with the average prediction among customers who were current and did not become delinquent. Given the difficulty of predicting default among individuals that currently show no sign of delinquency, we anticipate the model's performance to be less impressive than the values reported in Table 11. Nonetheless, the values reported in Table 12 indicate that the model is able to distinguish between these two populations. For instance, using input data from 2008Q4, the average model prediction for individuals who were current on their debts

Table 11: 1 Quarter Ahead Predictions, Full Sample– Hybrid DNN-GBT

Training Window	Testing Window	Data	Predicted	Delinquents	Non-Delinquents
2004Q1-2013Q4	2004Q1-2013Q4	0.3396	0.3359	0.7475	0.1242
2004Q1	2006Q1	0.3248	0.2948	0.6543	0.1218
2004Q2	2006Q2	0.3274	0.3057	0.6748	0.1260
2004Q3	2006Q3	0.3306	0.3126	0.6851	0.1286
2004Q4	2006Q4	0.3347	0.3153	0.6850	0.1293
2005Q1	2007Q1	0.3410	0.3185	0.6867	0.1279
2005Q2	2007Q2	0.3444	0.3224	0.6897	0.1295
2005Q3	2007Q3	0.3469	0.3224	0.6872	0.1287
2005Q4	2007Q4	0.3505	0.3306	0.6975	0.1327
2006Q1	2008Q1	0.3535	0.3390	0.7093	0.1366
2006Q2	2008Q2	0.3545	0.3364	0.7022	0.1355
2006Q3	2008Q3	0.3558	0.3369	0.7046	0.1338
2006Q4	2008Q4	0.3587	0.3434	0.7109	0.1379
2007Q1	2009Q1	0.3588	0.3504	0.7221	0.1425
2007Q2	2009Q2	0.3580	0.3528	0.7250	0.1452
2007Q3	2009Q3	0.3573	0.3550	0.7269	0.1482
2007Q4	2009Q4	0.3589	0.3571	0.7286	0.1492
2008Q1	2010Q1	0.3589	0.3606	0.7319	0.1527
2008Q2	2010Q2	0.3568	0.3633	0.7352	0.1570
2008Q3	2010Q3	0.3559	0.3632	0.7336	0.1586
2008Q4	2010Q4	0.3544	0.3636	0.7312	0.1618
2009Q1	2011Q1	0.3541	0.3614	0.7296	0.1595
2009Q2	2011Q2	0.3511	0.3567	0.7221	0.1590
2009Q3	2011Q3	0.3500	0.3557	0.7214	0.1588
2009Q4	2011Q4	0.3484	0.3536	0.7221	0.1565
2010Q1	2012Q1	0.3467	0.3567	0.7300	0.1585
2010Q2	2012Q2	0.3434	0.3518	0.7257	0.1563
2010Q3	2012Q3	0.3396	0.3517	0.7307	0.1568
2010Q4	2012Q4	0.3358	0.3484	0.7285	0.1562
2011Q1	2013Q1	0.3341	0.3479	0.7318	0.1553
2011Q2	2013Q2	0.3317	0.3436	0.7266	0.1536
2011Q3	2013Q3	0.3298	0.3426	0.7286	0.1527
2011Q4	2013Q4	0.3275	0.3402	0.7289	0.1509

Notes: Performance metrics for our model of default risk over 32+1 testing windows. For each testing window, the model is calibrated on data over the period specified in the training window, and predictions are based on the data available as of the data in the training window. For example, the fourth row reports the performance of the model calibrated using input data available in 2004Q3, and applied to 2006Q3 data to generate forecasts of delinquencies for within 8 quarter delinquencies. Average model forecasts over all customers, and customers that (ex-post) did and did not become 90+ days delinquent over the testing window are also reported. Source: Authors' calculations based on Experian Data.

and became 90+ days delinquent is 40.37%, contrasted with 10.53% for those who did not. As in Table 11, the model's ability to distinguish between these two classes is consistent across the 32+1 evaluation periods listed in Table 12.

Under certain conditions, the forecasts generated by our model can be converted to binary decisions by comparing the forecast to a specified threshold and classifying accounts with scores exceeding that threshold as high-risk. Setting the threshold level comes with a trade-off. A low level threshold leads to many accounts being classified as high risk, and even though this approach may accurately capture customers who are actually high-risk and about to default on their payments, it can also give rise to many low-risk accounts incorrectly classified as high-risk. By contrast, a high threshold can result in too many high-risk accounts being classified as low-risk.

This type of trade-off is inherent in any classification problem, and involves trading off Type-I (false positives) and Type-II (false negatives) errors in a classical hypothesis testing context. In the credit risk management context, a cost/benefit analysis can be formulated contrasting false positives to false negatives to make this trade-off explicit, and applying the threshold that will optimize an objective function in which costs and benefits associated with false positives and false negatives are inputs.

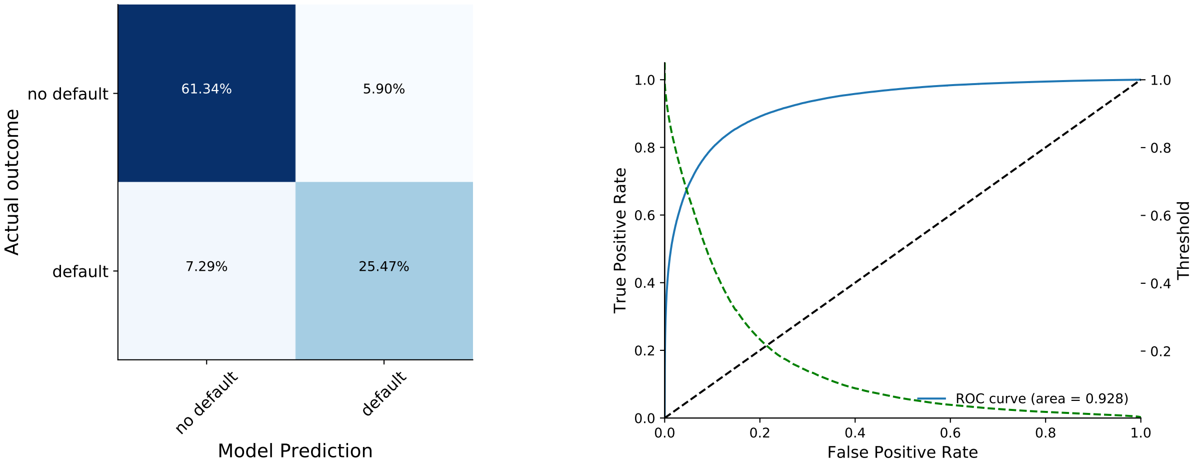
A commonly used performance metric in the machine learning and statistics literature is a 2×2 contingency table, often referred to as the confusion matrix, that describes the statistical behavior of any classification algorithm. In our application, the two rows correspond to ex post realizations of the two types of accounts in our sample, no default and default. We define no default accounts as those who do not become 90+ days delinquent during the forecast period, and default accounts as those who do. The two columns correspond to ex ante classifications of the accounts into these categories. If a predictive model is applied to a set of accounts, each account falls into one of the four cells in the confusion matrix, thus the performance of the model can be assessed by the relative frequencies of the entries. In the Neymann-Pearson hypothesis-testing framework, the lower-left entry is defined as Type-I error and the upper right as Type-II error, while the objective of the researcher is to minimize Type-II error (i.e., maximize "power") subject to a fixed level of Type-I error (i.e., "size").

As an illustration, Figure 8 Panel (a) shows the confusion matrix for our hybrid DNN-GBT model calibrated using 2011Q4 data and evaluated on 2013Q4 data and a threshold of 50%. This means that accounts with estimated delinquency probabilities greater than 50% are classified as default and 50% or below as no default. For this quarter, the model classified 61.34% + 7.29% = 68.63% of the accounts as no default, of which 61.34% did indeed not default and 7.29% actually defaulted, that is, they were 90+ days delinquent in

Table 12: 1 Quarter Ahead Predictions, Current– Hybrid DNN-GBT

Training Window	Testing Window	Data	Predicted	Delinquent	Non-delinquent
2004Q1-2013Q4	2004Q1-2013Q4	0.1676	0.1629	0.5304	0.0889
2004Q1	2006Q1	0.1844	0.1568	0.4292	0.0952
2004Q2	2006Q2	0.1702	0.1475	0.3972	0.0963
2004Q3	2006Q3	0.1695	0.1499	0.4007	0.0987
2004Q4	2006Q4	0.1727	0.1506	0.4010	0.0983
2005Q1	2007Q1	0.1805	0.1542	0.4047	0.0990
2005Q2	2007Q2	0.1813	0.1545	0.3994	0.1003
2005Q3	2007Q3	0.1831	0.1527	0.3929	0.0988
2005Q4	2007Q4	0.1847	0.1566	0.4032	0.1007
2006Q1	2008Q1	0.1890	0.1650	0.4195	0.1057
2006Q2	2008Q2	0.1896	0.1626	0.4098	0.1048
2006Q3	2008Q3	0.1872	0.1593	0.4043	0.1028
2006Q4	2008Q4	0.1817	0.1595	0.4037	0.1053
2007Q1	2009Q1	0.1781	0.1650	0.4205	0.1097
2007Q2	2009Q2	0.1752	0.1668	0.4240	0.1122
2007Q3	2009Q3	0.1713	0.1689	0.4302	0.1149
2007Q4	2009Q4	0.1661	0.1669	0.4230	0.1160
2008Q1	2010Q1	0.1683	0.1722	0.4372	0.1186
2008Q2	2010Q2	0.1668	0.1778	0.4508	0.1231
2008Q3	2010Q3	0.1661	0.1795	0.4559	0.1244
2008Q4	2010Q4	0.1644	0.1787	0.4509	0.1252
2009Q1	2011Q1	0.1674	0.1812	0.4616	0.1248
2009Q2	2011Q2	0.1668	0.1768	0.4514	0.1218
2009Q3	2011Q3	0.1669	0.1769	0.4520	0.1218
2009Q4	2011Q4	0.1597	0.1699	0.4380	0.1189
2010Q1	2012Q1	0.1604	0.1724	0.4468	0.1200
2010Q2	2012Q2	0.1622	0.1705	0.4477	0.1168
2010Q3	2012Q3	0.1598	0.1676	0.4434	0.1152
2010Q4	2012Q4	0.1575	0.1668	0.4432	0.1152
2011Q1	2013Q1	0.1606	0.1710	0.4576	0.1162
2011Q2	2013Q2	0.1603	0.1692	0.4541	0.1149
2011Q3	2013Q3	0.1578	0.1660	0.4496	0.1128
2011Q4	2013Q4	0.1548	0.1623	0.4430	0.1109

Notes: Performance metrics for our model of default risk over 32+1 testing windows for customers who are current as of the forecast date but become 90+ days delinquent in the following 8 quarters. For each testing window, the model is calibrated on data over the period specified in the training window columns, and predictions are based on the data available as of the data in the training window. For example, the fourth row reports the performance of the model calibrated using input data available in 2004Q3, and applied to 2006Q3 data to generate forecasts of delinquencies for within 8 quarter delinquencies. Average model forecasts over all current customers, and all current customers that did and did not become 90+ days delinquent over the testing window are also reported. Source: Authors' calculations based on Experian Data.



(a) Confusion Matrix

(b) ROC Curve

Figure 8: Confusion Matrix and Receiver Operating Characteristic (ROC) Curve

Notes: Confusion matrix and Receiver Operating Characteristic (ROC) curve of out-of-sample forecasts of 90+ days delinquencies over the 8Q forecast horizon based on our model of default risk. In Panel (a), rows correspond to actual states, with default defined as 90+ days delinquent, no default otherwise. Classifier threshold: 50%. The numerical example is based on the model calibrated on 2011Q4 data and applied to 2013Q4 to generate out-of-sample predictions. Source: Authors' calculations based on Experian Data.

the subsequent 8 quarters. By the same token, of the 5.9% + 25.47% = 31.37% borrowers who defaulted, the model accurately classified 25.47%. Thus, the model's accuracy, defined as the percent of instances correctly classified, is the sum of the entries on the diagonal of the confusion matrix, that is, 61.34 % + 25.47% = 86.81%.

We can compute three additional performance metrics from the entries of the confusion matrix, which we describe heuristically here and define formally in the appendix. Precision measures the model's accuracy in instances that are classified as default. Recall refers to the number of accounts that defaulted as identified by the model divided by the actual number of defaulting accounts. Finally, the F-measure is simply the harmonic mean of precision and recall. In an ideal scenario, we would have very high precision and recall.

We can track the trade-off between true and false positives by varying the classification threshold of our model, and this trade-off is plotted in Figure 8 Panel (b). The blue line, called the Receiver Operating Characteristic (ROC) curve, is the pairwise plot of true and false positive rates for different classification thresholds (green line), and as the threshold decreases, the figure shows that the true positive rate increases, but so does the false positive rate. The ROC curve illustrates the non-linear nature of the trade-offs, implying that increase in true positive rates is not always proportionate with the increase in false positive rates. The optimal threshold then considers the cost of false positives with respect to the gain of true positives. If these are equal, the optimal threshold will correspond to the tangent point of the ROC curve with the 45 degree line.

The last performance metric we consider is the area under the ROC curve, known as AUC score, which is a widely used measure in the machine-learning literature for comparing models. It can be interpreted as the probability of the classifier assigning a higher probability of being in default to an account that is actually in default. The ROC area of our model ranges from 0.9238 to 0.9300, demonstrating that our machine-learning classifiers have strong predictive power in separating the two classes.

Table 13 reports the performance metrics widely used in the machine-learning literature for each of the 32+1 models discussed. Our models exhibit strong predictive power across the various performance metrics. For instance, the 85.71% precision implies that when our classifier predicts that someone is going to default, there is an 85.71% chance this person will actually default; while the 72.67% recall means that we accurately identified 72.67% of all the defaulters. Our approach of using only one quarter of data to train the model is rather restrictive. Using more quarters usually increases model performance, so since most credit scoring applications will use a training data that exceeds one quarter, performance metrics are likely to improve relative to what we report in our exercise.

Table 14 reports the same performance metrics for the population of borrowers who are current, that is, they do not have any delinquencies in the quarter they are assessed. As previously noted, this is a smaller population with a lower probability of default. Performance metrics drop marginally relative to those for the model applied to the population of all borrowers but they are still very strong. For example, the AUC score drops from 92-93% to 86-88%, accuracy and loss mostly remain in the same range.

A.4 Model Interpretation

We use our hybrid DNN-GBT model to uncover associations between the explanatory variables and default behavior. Since we do not identify causal relationships, our goal is simply to find covariates that have an important impact on default outcomes. Our findings can be used to better understand default behavior, further refine model specification and possibly aid in the formulation of theoretical models of consumer default. For this exercise, we mainly use the pooled model, which uses all available data. This allows us to assess factors that are critical in default behavior throughout the sample period with the best performing model. We also consider time variation in the factors influencing the default decision in subsets of our sample.

Table 13: Performance Metrics using Hybrid DNN-GBT, Full Sample

Training Window	Testing Window	AUC score	Precision	Recall	F-measure	Accuracy	Loss
2004Q1-2013Q4	2004Q1-2013Q4	0.9494	0.8546	0.8104	0.8319	0.8888	0.2693
2004Q1	2006Q1	0.9243	0.8508	0.7056	0.7714	0.8642	0.3230
2004Q2	2006Q2	0.9250	0.8486	0.7170	0.7773	0.8654	0.3187
2004Q3	2006Q3	0.9259	0.8492	0.7261	0.7828	0.8668	0.3165
2004Q4	2006Q4	0.9250	0.8517	0.7232	0.7822	0.8652	0.3200
2005Q1	2007Q1	0.9257	0.8561	0.7235	0.7842	0.8642	0.3208
2005Q2	2007Q2	0.9257	0.8571	0.7267	0.7865	0.8641	0.3217
2005Q3	2007Q3	0.9251	0.8592	0.7224	0.7849	0.8626	0.3247
2005Q4	2007Q4	0.9238	0.8541	0.7290	0.7866	0.8614	0.3278
2006Q1	2008Q1	0.9246	0.8505	0.7390	0.7908	0.8618	0.3265
2006Q2	2008Q2	0.9245	0.8542	0.7319	0.7883	0.8607	0.3275
2006Q3	2008Q3	0.9255	0.8556	0.7342	0.7902	0.8613	0.3259
2006Q4	2008Q4	0.9257	0.8529	0.7402	0.7926	0.8610	0.3259
2007Q1	2009Q1	0.9277	0.8487	0.7540	0.7986	0.8635	0.3210
2007Q2	2009Q2	0.9279	0.8441	0.7614	0.8006	0.8642	0.3198
2007Q3	2009Q3	0.9286	0.8422	0.7673	0.8030	0.8655	0.3177
2007Q4	2009Q4	0.9300	0.8454	0.7723	0.8072	0.8676	0.3143
2008Q1	2010Q1	0.9299	0.8415	0.7787	0.8089	0.8679	0.3153
2008Q2	2010Q2	0.9296	0.8334	0.7858	0.8089	0.8675	0.3159
2008Q3	2010Q3	0.9292	0.8323	0.7864	0.8087	0.8676	0.3161
2008Q4	2010Q4	0.9290	0.8297	0.7850	0.8067	0.8667	0.3171
2009Q1	2011Q1	0.9296	0.8341	0.7842	0.8084	0.8683	0.3154
2009Q2	2011Q2	0.9282	0.8341	0.7748	0.8033	0.8668	0.3179
2009Q3	2011Q3	0.9284	0.8379	0.7721	0.8036	0.8679	0.3168
2009Q4	2011Q4	0.9288	0.8378	0.7702	0.8026	0.8680	0.3154
2010Q1	2012Q1	0.9293	0.8323	0.7770	0.8037	0.8684	0.3142
2010Q2	2012Q2	0.9280	0.8290	0.7731	0.8001	0.8673	0.3162
2010Q3	2012Q3	0.9277	0.8248	0.7746	0.7989	0.8676	0.3151
2010Q4	2012Q4	0.9271	0.8172	0.7769	0.7965	0.8667	0.3167
2011Q1	2013Q1	0.9280	0.8160	0.7790	0.7971	0.8675	0.3141
2011Q2	2013Q2	0.9276	0.8158	0.7754	0.7951	0.8674	0.3139
2011Q3	2013Q3	0.9281	0.8127	0.7792	0.7956	0.8680	0.3123
2011Q4	2013Q4	0.9284	0.8118	0.7776	0.7943	0.8681	0.3104

Notes: Performance metrics for our model of default risk. The model calibrations are specified by the training and testing windows. The results of classifications versus actual outcomes over the following 8Q are used to calculate these performance metrics for 90+ days delinquencies within 8Q. Source: Authors' calculations based on Experian Data.

Table 14: Performance Metrics using Hybrid DNN-GBT, Current

Training Window	Testing Window	AUC score	Precision	Recall	F-measure	Accuracy	Loss
2004Q1-2013Q4	2004Q1-2013Q4	0.9225	0.781	0.5632	0.6545	0.9003	0.2458
2004Q1	2006Q1	0.8773	0.7686	0.4059	0.5313	0.8679	0.3165
2004Q2	2006Q2	0.8653	0.7298	0.3569	0.4794	0.8681	0.3159
2004Q3	2006Q3	0.8638	0.7227	0.3606	0.4811	0.8681	0.3162
2004Q4	2006Q4	0.8642	0.7307	0.3609	0.4832	0.8666	0.3196
2005Q1	2007Q1	0.8649	0.7384	0.3658	0.4892	0.8621	0.3275
2005Q2	2007Q2	0.8621	0.7302	0.3591	0.4814	0.8597	0.3318
2005Q3	2007Q3	0.8616	0.7382	0.3454	0.4706	0.8577	0.3353
2005Q4	2007Q4	0.8599	0.7261	0.3594	0.4808	0.8566	0.3382
2006Q1	2008Q1	0.8612	0.7192	0.3818	0.4988	0.8550	0.3402
2006Q2	2008Q2	0.8615	0.7236	0.3656	0.4858	0.8532	0.3414
2006Q3	2008Q3	0.8609	0.7219	0.3590	0.4795	0.8541	0.3403
2006Q4	2008Q4	0.8578	0.7061	0.3552	0.4727	0.8560	0.3366
2007Q1	2009Q1	0.8599	0.6993	0.3818	0.4939	0.8607	0.3285
2007Q2	2009Q2	0.8596	0.6870	0.3903	0.4978	0.8621	0.3251
2007Q3	2009Q3	0.8594	0.6802	0.3990	0.5030	0.8649	0.3207
2007Q4	2009Q4	0.8573	0.6747	0.3898	0.4941	0.8674	0.3170
2008Q1	2010Q1	0.8607	0.6822	0.4156	0.5165	0.8690	0.3152
2008Q2	2010Q2	0.8617	0.6645	0.4385	0.5284	0.8694	0.3136
2008Q3	2010Q3	0.8619	0.6608	0.4480	0.5340	0.8701	0.3128
2008Q4	2010Q4	0.8632	0.6653	0.4388	0.5288	0.8715	0.3098
2009Q1	2011Q1	0.8664	0.6769	0.4524	0.5423	0.8721	0.3092
2009Q2	2011Q2	0.8666	0.6833	0.4386	0.5342	0.8724	0.3085
2009Q3	2011Q3	0.8678	0.6891	0.4373	0.5351	0.8732	0.3076
2009Q4	2011Q4	0.8651	0.6866	0.4123	0.5152	0.8760	0.3029
2010Q1	2012Q1	0.8664	0.6781	0.4255	0.5229	0.8754	0.3029
2010Q2	2012Q2	0.8684	0.6851	0.4265	0.5257	0.8752	0.3025
2010Q3	2012Q3	0.8673	0.6854	0.4178	0.5192	0.8763	0.3012
2010Q4	2012Q4	0.8664	0.6768	0.4195	0.5180	0.8770	0.2998
2011Q1	2013Q1	0.8691	0.6744	0.4420	0.5340	0.8761	0.3005
2011Q2	2013Q2	0.8694	0.6781	0.4372	0.5317	0.8765	0.2997
2011Q3	2013Q3	0.8690	0.6766	0.4336	0.5285	0.8779	0.2975
2011Q4	2013Q4	0.8671	0.6706	0.4270	0.5218	0.8789	0.2958

Notes: Performance metrics for our model of default risk for the current population. Borrowers who are current do not have any delinquencies. The model calibrations are specified by the training and testing windows. The results of classifications versus actual outcomes over the following 8Q are used to calculate these performance metrics for 90+ days delinquencies within 8Q. Source: Authors' calculations based on Experian Data.

A.4.1 Explanatory Power of Variables

We start by examining the explanatory power of each of our features. We follow an approach similar to Sirignano, Sadhwani, and Giesecke (2018), which amounts to a perturbation analysis on the pooled sample using our hybrid model. First, we draw a random sample of 100,000 observations from the testing sample. Then, for each variable, we re-shuffle the feature, keeping the distribution intact and the model's loss function is evaluated with the changed covariate. We repeat this step 10 times, and report the average of the loss and accuracy. Then, the variable is replaced to its original values, and a perturbation test is performed on a new variable. Perturbing the variable of course reduces the accuracy of the model, and the test loss becomes larger. If a particular variable has strong explanatory power, the test loss will significantly increase. The test loss for the complete model when no variables are perturbed is the Baseline value. Features that have large explanatory power, and whose information is not contained in the other remaining variables will increase the loss significantly if they are altered. Table 15 reports the results. Features relating to credit history, debt balances, and the number and credit available on revolving trades dominate the list. Specifically, credit amount on open trades increases the loss by 24%, debt balances by 17%, the number of open credit and bankcards by 15%, while the total monthly payment on open trades, months since oldest trade, and months since the most recent 90+ days delinquency each increase the loss by 11%. These results suggest that debt balances, length of credit history and temporal proximity to a delinquency are all important factors in default behavior. Based on publicly available information, length of the credit history is also an important determinant of standard credit scoring models, though payment history rather than balances or number of trades is understood as the most critical.

This approach to assessing the importance of different features for the predicted probability of default has two major shortcomings. First, when features are highly correlated, the interpretation of feature importance can be biased by unrealistic data instances. To illustrate this problem, consider two highly correlated features. As we perturb one of the features, we create instances that are unlikely or even impossible. For example, mortgage balances are highly correlated with and lower than total debt balances, yet this perturbation approach could create instances in which total debt balances are smaller than mortgage balances. Since many of the features are strongly correlated, care must be taken with interpretation of feature importance. We list the highly correlated features in Appendix A. An additional concern with this perturbation approach is that the distribution of some features are highly skewed, which implies that the probability of their value being different than where the mass of their distribution is concentrated is quite low. Moreover, skewness varies substantially across features, therefore the informativeness of the perturbation may differ across variables.

In the next section, we examine a more robust approach that is less susceptible to these limitations.

Table 15: Explanatory Power of Variables

Feature	Accuracy	Loss
Credit amount on open trades	0.8587	0.3333
Total debt balances	0.8627	0.3245
Number of credit & bankcards	0.8677	0.3156
Credit amount on open credit card trades	0.8714	0.3064
Monthly payment on open trades	0.8751	0.3039
Credit amount on open revolving trades	0.8751	0.3036
Months since the most recent 90 or more days delinquency	0.8757	0.3034
Months since the oldest trade was opened	0.8722	0.3026
Balance on credit & bankcards	0.8751	0.2974
Number of open mortgage type trades	0.8758	0.2953
Monthly payment on credit card trades	0.8777	0.2946
Credit amount on open joint revolving trades	0.8811	0.2930
Monthly payment on open first mortgage trades	0.8817	0.2876
Number of installment trades	0.8790	0.2874
Months since the most recent 30-180days delinquency on trades	0.8806	0.2861
Worst ever status on a trade in the last 24 months	0.8797	0.2859
Balance on installment trades	0.8802	0.2859
Joint debt balances	0.8818	0.2856
Installment utilization	0.8804	0.2853
Mortgage to total debt	0.8817	0.2846
Number of open auto loan trades	0.8803	0.2843
Months since the most recently closed, transferred, or refinanced first mortgage trade	0.8809	0.2835
Months since the most recently opened home equity line of credit trade	0.8816	0.2835
Months since the most recent 30-180days delinquency on credit card trades	0.8800	0.2835
Balance on open auto loan trades	0.8812	0.2826
Inquiries made in the last 12 months	0.8818	0.2826
Months since the most recently opened first mortgage trade	0.8819	0.2822
Unsatisfied collections	0.8832	0.2814
⋮	⋮	⋮
Mortgage type inquiries made in the last 3 months	0.8879	0.2694
Bankcard revolving and charge inquiries made in the last 3 months	0.8881	0.2693
Baseline	0.8881	0.2691

Notes: This table reports a perturbation analysis on the pooled sample using our hybrid model. For each variable, we re-shuffle the feature, keeping the distribution intact in the test dataset and the model's loss function is evaluated on the test dataset with the changed covariate. We repeat this step 10 times, and report the average of the loss and accuracy. Then, the variable is replaced to its original values, and a perturbation test is performed on a new variable. Perturbing the variable of course reduces the accuracy of the model, and the test loss becomes larger. If a particular variable has strong explanatory power, the test loss will significantly increase. The test loss for the complete model when no variables are perturbed is the Baseline value.

A.4.2 Economic Significance of Variables

We now turn to analyzing the economic significance of our features for default behavior. We adopt SHapley Additive exPlanations (SHAP), a unified framework for interpreting predictions, to explain the output of our hybrid deep learning model (for a detailed description of the approach see Lundberg and Lee (2017)). SHAP uses a game theoretical concept to assign each feature a local importance value for a given prediction. Though Shapley values are local by design, they can be combined into global explanations by averaging the absolute

Shapley values featurewise. Then, we can compare features based on their absolute average Shapley values, with higher values implying higher feature importance. Similarly to permutation feature importance, SHAP is a feature importance measure. The main difference between the two is that while permutation feature importance is based on the decrease in model performance, SHAP is based on the magnitude of feature attributions.

We first compute the Shapley values for the Deep Neural Network model and the Gradient Boosted Trees model separately, then simply average them for each individual and for each feature.[19] We use a random sample of 100,000 observations for explaining the model. By the Shapley efficient property, the SHAP values for an observation sum up to the difference between the predicted value of that observation and the expected value, computed using the background dataset:

$$f(x) = E_X[\widehat{f}(X)] + \sum_{j=1}^{M} \phi_j \tag{15}$$

where f is the model prediction, M is the number of features, and $\phi_j \in R$ is the feature attribution for feature j (i.e., the Shapley values). Thus, we can interpret the Shapley value as the contribution of a feature value to the difference between the model's prediction and the mean prediction, given the current set of feature values. As an illustration, a SHAP value of 0.1 implies that the feature's value for that particular instance contributed to an increase of 0.1 to the predicted probability compared to the mean prediction. Features that are highly correlated can decrease the importance of the associated feature by splitting the importance between both features. We account for the effect of feature correlation on interpretability by grouping features with a correlation larger than 0.7, and summing the SHAP values within each groups. We denote these groups with an asterisk for the rest of the analysis and report the composition of feature groups in Table 19 in the appendix.

Figure 9 sorts features by the sum of absolute SHAP value magnitudes, and plots the distribution of the impact each feature has on the model output for the twelve most important features or groups of correlated features. The color represents the feature value (red: high, blue: low), whereas the position on the horizontal axis denotes the contribution of the feature. The charts plot the distribution of SHAP values for individual instances in the 100,000 testing sample. The most important feature in terms of SHAP value magnitude is the worst status on any trades. High values of this variable tend to increase predicted default risk, whereas low values tend to decrease it, though the distribution of instances

[19] We implement Deep SHAP, a high-speed approximation algorithm for SHAP values in deep learning models to compute the Shapley values for our 5 hidden layer neural network. For GBT, we implement TreeExplainer, a high-speed exact algorithm for tree ensemble methods. Because our dataset is fairly large with many features, we pass a random sample of 100 observations, referred to as background observations, to compute the expected value for both models.

is dispersed. Features capturing credit history, such as length of credit history and recent delinquencies, also have high SHAP values, specifically, high values of these variables lower predicted default risk, with a much more dispersed distribution. Additionally, delinquent balances and outstanding collections are typically associated with an increase in predicted default probability. Higher total debt balances are also associated with a lower than expected predicted default risk, reiterating the notion that the borrowers with the most credit are also associated with lower predicted probability of default, which suggests that credit allocation decisions are made to minimize default probabilities. As in the perturbation exercise, we find that number of trades and balances seem to have the strongest association with variation in the predicted probability of default, whereas credit inquiries do not play a sizable role.

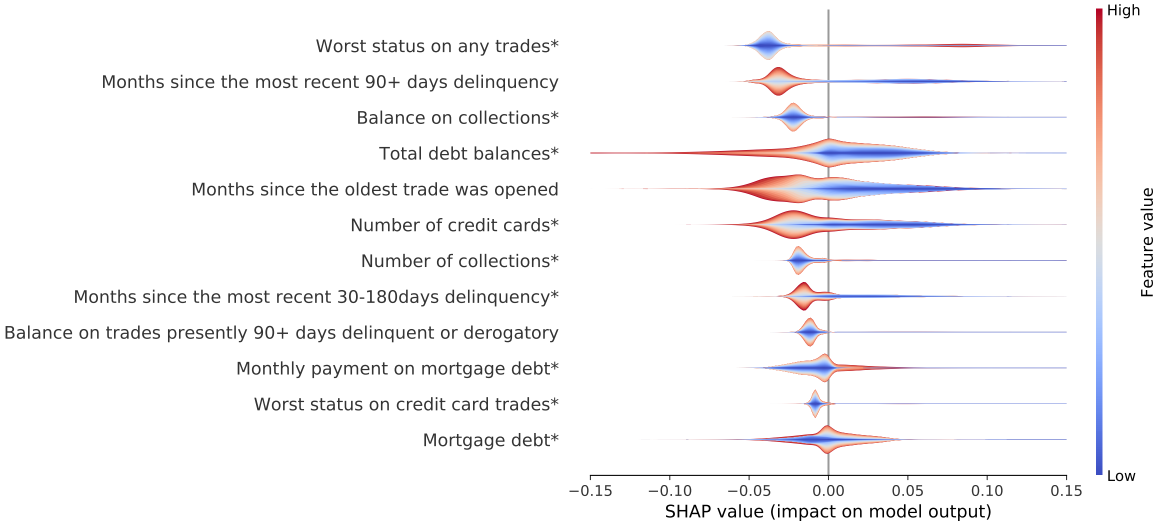


Figure 9: SHAP applied to predicted 90+ days delinquency within 8Q

Notes: Source: Authors' calculations based on Experian Data.

These results only point to correlations between the features and the predicted outcome and should not be interpreted causally. Yet, they can be used as a point of departure for a causal analysis of default and theoretical modeling. They are also important to comply with legal disclosure requirements. Both the Fair Credit Reporting Act ad the Equal Opportunity in Credit Access Act require lenders and developers of credit scoring models to reveal the most important factors leading to a denial of a credit application and for credit scores. The SHAP value provides an individualized assessment of such factors that can be used for making credit allocation decisions and communicating them to the borrower.

A.4.3 Temporal Determinants of Default

We next look at the changing dynamics of default behavior by comparing models that are trained in different periods of time. For this analysis, we use our hybrid model. Specifically, we target the following time periods: 2006Q1, 2008Q1 and 2011Q1 as time periods before, during and after the 2007-2009 crisis, and compute default predictions for them with data trained in the same quarter and two years prior, that is in 2004Q1, 2006Q1, 2007Q1 and 2009Q1, respectively. We then calculate Shapley values for the two models.[20] The first exercise provides an in-sample assessment for feature importance, while the second exercise can be used to assess feature importance out-of-sample. In both exercises, the model is the same, so comparing the results from the two exercises can help uncover which features are important for default prediction for a given period from an ex ante perspective and from an in-sample perspective. Table 16 reports the results.[21] For each period, it is interesting to compare the variation in SHAP values from an ex ante and contemporaneous perspective, and additionally we are interested in comparing variation in SHAP values for given features in the different time periods. In most testing windows, the temporal proximity to delinquency has the highest SHAP value.

Debt balances (i.e, total debt, revolving debt, auto debt), and utilization on installment loans are consistently among the five most influential features. Mortgage debt and credit card debt, are generally between the fifth and twenty-first most significant in terms of SHAP values. The SHAP value is quite stable over time for most features, but there are some variables for which it changes substantially. One example is number of open credit cards which ranks second and seventh for 2004Q1 and 2006Q1 but moves down to fourteenth in sample and nineteenth out-of sample for 2011Q1. The length of the credit history is never among the thirty most important features. Overall these results confirm our findings from the pooled model, suggesting the balances and number of trades, in addition to delinquency status, have a strong association with default risk according to our model.

A.5 Comparison with Credit Scores

The credit score is a summary indicator intended to predict the risk of default by the borrower and it is widely used by the financial industry. For most unsecured debt, lenders typically verify a perspective borrower's credit score at the time of application and sometimes a short

[20] We do this for both the Deep Neural Network and the Gradient Boosted Trees and similarly to how we obtain the output, we simply take the average of the Shapley values. For both our models, we use a random sample of 100 observations of the testing data scaled by the mean and standard deviation of the corresponding training data for reference value.

[21] The features are sorted by the sum of absolute SHAP value magnitudes over the first period.

Table 16: SHAP Values over Time

	Prediction Date					
	2006Q1		2008Q1		2011Q1	
	Model					
	2004Q1	2006Q1	2006Q1	2008Q1	2009Q1	2011Q1
Features						
Ratio of inquiries to trades opened in the last 6 months	0.035 (1)	0.029 (3)	0.026 (3)	0.027 (4)	0.029 (5)	0.023 (4)
Number of credit cards*	0.032 (2)	0.014 (7)	0.013 (7)	0.007 (17)	0.007 (19)	0.013 (14)
Total debt balances*	0.03 (3)	0.022 (5)	0.021 (5)	0.014 (8)	0.016 (10)	0.027 (3)
Balance on revolving debt*	0.029 (4)	0.041 (2)	0.034 (2)	0.031 (3)	0.032 (4)	0.013 (15)
Balance on auto loans*	0.028 (5)	0.02 (6)	0.018 (6)	0.015 (7)	0.028 (6)	0.015 (8)
Installment utilization	0.027 (6)	0.025 (4)	0.024 (4)	0.044 (2)	0.049 (2)	0.055 (1)
Worst status on any trades*	0.02 (7)	0.008 (10)	0.007 (8)	0.017 (5)	0.011 (14)	0.015 (9)
Months since the most recently opened home equity line of credit trade	0.018 (8)	0.004 (24)	0.004 (19)	0.003 (33)	0.003 (37)	0.005 (20)
Months since the most recent 90+ days delinquency	0.015 (9)	0.052 (1)	0.043 (1)	0.044 (1)	0.07 (1)	0.049 (2)
Monthly payment on credit card trades	0.014 (10)	0.002 (35)	0.002 (33)	0.002 (39)	0.004 (29)	0.01 (18)
Credit card utilization	0.014 (11)	0.008 (8)	0.007 (9)	0.01 (11)	0.019 (8)	0.021 (5)
Mortgage debt*	0.012 (12)	0.005 (16)	0.004 (20)	0.005 (21)	0.015 (12)	0.017 (7)
Heloc utilization	0.011 (13)	0.0 (64)	0.0 (64)	0.0 (64)	0.0 (64)	0.0 (64)
Number of bankcard revolving and charge inquiries	0.009 (14)	0.002 (36)	0.002 (38)	0.016 (6)	0.01 (15)	0.004 (23)
Worst status on credit card trades*	0.009 (15)	0.003 (28)	0.003 (31)	0.002 (37)	0.004 (27)	0.007 (19)
Balance on credit & bankcards	0.009 (16)	0.004 (19)	0.005 (15)	0.013 (10)	0.006 (24)	0.018 (6)
Number of open installment trades	0.005 (17)	0.008 (9)	0.007 (10)	0.006 (19)	0.004 (30)	0.005 (21)
Number of mortgage type inquiries made in the last 3 months	0.005 (18)	0.002 (41)	0.002 (37)	0.001 (43)	0.002 (41)	0.0 (56)
Balance on installment loans*	0.005 (19)	0.005 (15)	0.006 (11)	0.009 (14)	0.041 (3)	0.012 (16)
Balance on trades presently 90+ days delinquent	0.005 (20)	0.002 (33)	0.002 (36)	0.004 (25)	0.003 (32)	0.001 (43)

Notes: This table reports the Shapley values for a selected 20 features for three out-of-sample models. For each prediction window, we compute the Shapley value for each of the observations and for each feature. We then calculate the average of the absolute value for each feature, and report the results for the selected features. Finally, we rank the results based on the feature's relative rank in the given prediction window in parentheses. Source: Authors' calculations based on Experian data.

recent sample of their credit history. For larger unsecured debts, lenders also typically require some form of income verification, as they do for secured debts, such as mortgages and auto loans. Still, the credit score is often a key determinant of crucial terms of the borrowing contract, such as the interest rate, the downpayment or the credit limit.

The most widely known credit score is the FICO score, a measure generated by the Fair Isaac Corporation, which has been in existence in its current form since 1989. Each of the three major credit reporting bureaus– Equifax, Experian and TransUnion– also have their own proprietary credit scores. Credit scoring models are not public, though they are restricted by the law, mainly the Fair Credit Reporting Act of 1970 and the Consumer Credit Reporting Reform Act of 1996. The legislation mandates that consumers be made aware of the 4 main factors that may affect their credit score. Based on available descriptive materials from FICO and the credit bureaus, these are payment history and outstanding debt, which account for more than 65% of the variation in credit scores, followed by credit history, or the age of existing accounts, which explains 15% of the variation, followed by new accounts and types of credit used (10%) and new "hard" inquiries, that is credit report inquiries coming from prospective lenders after a borrower initiated credit application.

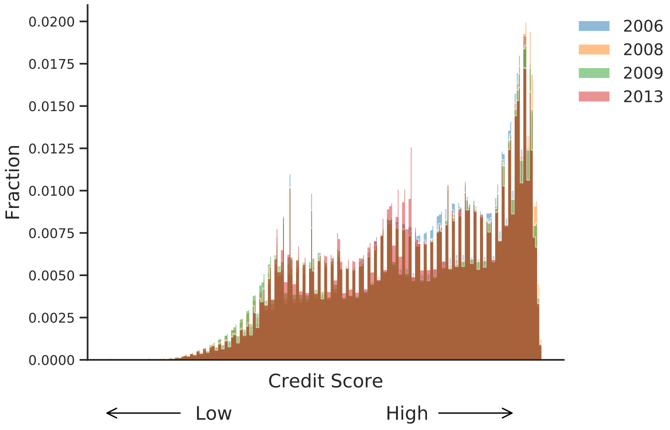


Figure 10: Credit Score Histogram by Years

Notes: Histogram of the credit score in our data by year for selected years. Source: Authors' calculations based on Experian Data.

U.S. law prohibits credit scoring models from considering a borrower's race, color, religion, national origin, sex and marital status, age, address, as well as any receipt of public assistance, or the exercise of any consumer right under the Consumer Credit Protection Act. The credit score cannot be based on information not found in a borrower's credit report, such as salary, occupation, title, employer, date employed or employment history, or interest rates being charged on particular accounts. Finally, any items in the credit report reported as child/family support obligations are not permitted, as well as "soft" inquiries[22] and any information that is not proven to be predictive of future credit performance.

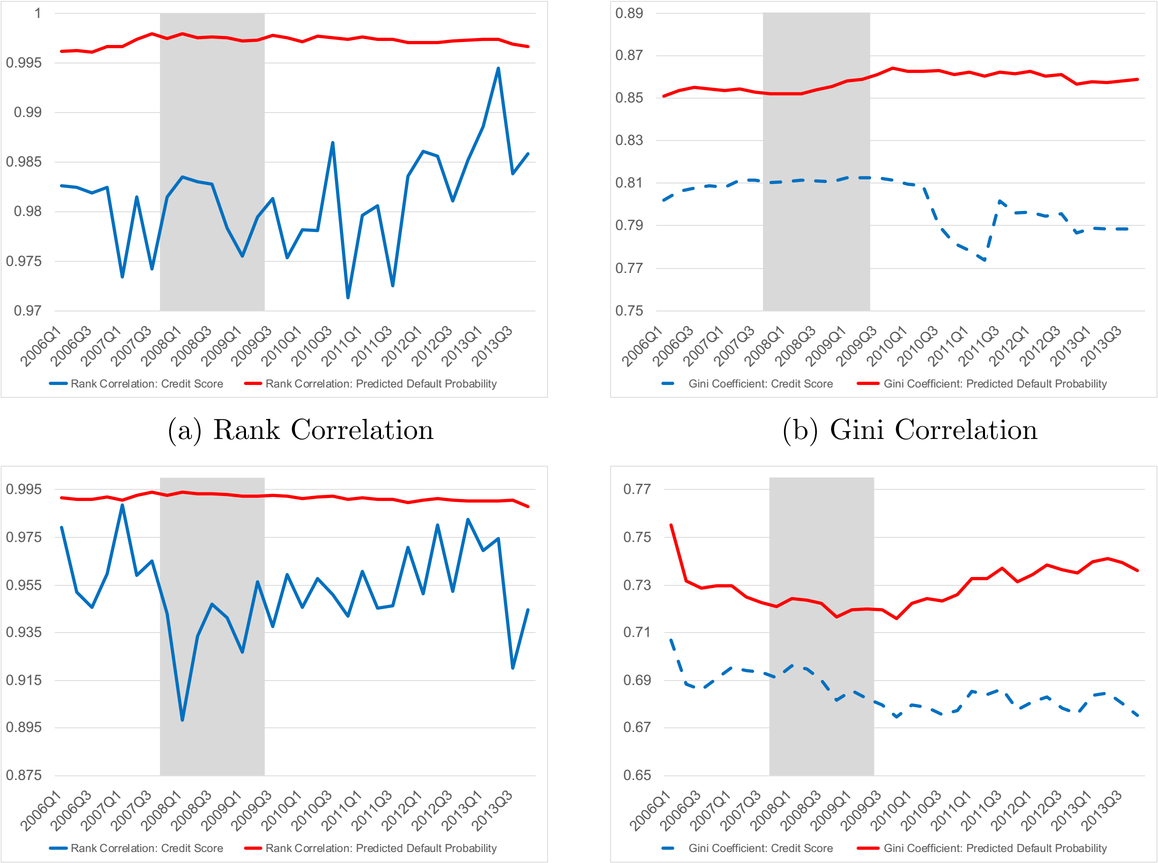
A.6 Additional Figures and Tables

Table 17: Distribution of Customers by Credit Score and Predicted Default

		Credit Score			
		Subprime, Near Prime	Prime Low	Prime Mid	Prime High, Superprime
Predicted Default bin	1	36.2%	3.6%	1.2%	0.4%
	2	3.7%	3.5%	2.4%	1.1%
	3	1.1%	2.7%	4.0%	3.3%
	4	0.3%	1.0%	3.5%	32.0%

Notes: This table reports the share of customers in each predicted credit score categories and corresponding predicted default probability bins. Customers are classified based on credit scores and predicted default probabilities at account origination for each of their credit cards included in the balances. Source: Authors' calculations based on Experian data.

[22] These include "consumer-initiated" inquiries, such as requests to view one's own credit report, "promotional inquiries", requests made by lenders in order to make pre-approved credit offers, or "administrative inquiries", requests made by lenders to review open accounts. Requests that are marked as coming from employers are also not counted.



(a) Rank Correlation

(b) Gini Correlation

(c) Rank Correlation: Current

(d) Gini Correlation: Current

Figure 11: Absolute Value of Rank Correlation with Realized Default Rate

Notes: Absolute value of rank correlation with realized default rate for the credit score and model predicted default probability for the full sample (a), for the current population (c), and Gini coefficients for the credit score and model predicted default probability by quarter for the full sample (b), and for the current population (d). Source: Authors' calculations based on Experian data.

Online Appendix to Predicting Consumer Default: A Deep Learning Approach

A Data Pre-Processing

A.1 Sample Restrictions

Our original dataset contains 33,600,000 observations. We discard observations of individuals with missing birth information, deceased individuals and restrict our analysis to individuals aged between 18 and 85, residing in one of the 50 states or the District of Columbia, with 8 consecutive quarters of non-missing default behavior. This leaves us with 22,004,753 data points. Our itemized sample restrictions are summarized in Table 18 below.

Table 18: Itemized Sample Restrictions

	Observations
Credit Report Data	33,600,000
Remove	
Deceased	- 513,270
Age	- 4,718,804
Residence	- 953,215
Prediction Window	- 5,409,958
Prediction Sample	22,004,753

A.2 Feature Scaling

We normalize all explanatory variables by their means and standard deviations:

$$z_i = \frac{x_i - \mu_x}{\sigma_x} \tag{16}$$

where $\mathbf{x} = (x_1, x_2, \dots, x_k)$, and $\mathbf{z_i}$ is the i^{th} normalized data.

A.3 Feature Groups

For the SHAP value analysis, we grouped features that had a correlation higher than 0.7. These groups are presented in Table 19.

Table 19: Feature Groups

Total debt balances*
Joint debt balances
Credit amount on joint trades
Total debt balances
Credit amount on open trades

Balance on revolving debt*
Credit amount on home equity line of credit trades
Balance on home equity line of credit trades
Balance on joint revolving trades
Credit amount on joint revolving trades
Credit amount on revolving trades

Balance on auto loans*
Balance on open auto loan trades
Number of auto loan trades

Balance on collections*
Balance on collections
Balance on collections, placed in the last 24 months

Balance on installment loans*
Balance on installment trades
Balance on joint installment trades

Number of bankruptcies*
Public record bankruptcies
Public record discharged bankruptcies

Number of collections*
Number of collections
Unsatisfied collections

Number of credit cards*
Credit amount on open credit card trades
Number of credit & bankcards

Monthly payment on mortgage debt*
Monthly payment on joint mortgage type trades
Monthly payment on first mortgage trades

Monthly payment on debt*
Monthly payment on debt
Monthly payment on joint installment trades

Balance on 90-180 days late installment loans*
Amount past due on installment trades presently 90+ dpd
Balance on installment trades presently 90+ dpd

Fraction of 90+ days delinquent debt*
Fraction of 90+ days delinquent debt
Worst present status on an open trade

90 days late credit card debt*
Amount past due on credit card trades presently 90+ dpd
Balance on bankcard revolving and charge trades presently 90+ dpd

Months since the most recent 30-180days delinquency*
Months since the most recent 30-180 days delinquency on credit card trades
Months since the most recent 30-180 days delinquency

Worst status on credit card trades*
Worst present status on a trade (excluding collections)
Worst present status on a credit card trade

Worst status on any trades*
Worst present status on a trade
Worst ever status on a trade in the last 24 months

Mortgage debt*
Mortgage to total debt
Number of open mortgage type trades
Months since the most recently opened first mortgage trade
Months since the most recently closed, transferred or refinanced first mortgage trade

A.4 Train-Test Split

For most of our analysis we split the data to account for look-ahead bias, i.e., the training set consists of data 8Q prior to the testing data. Then, we scale the testing data by the mean and standard deviation of the training data. In an alternative specification, we split our pooled data into three chunks: training set (60%), holdout set (20%), and testing set (20%). We report each specifications in Table 11 - Table 12. Except for parts of Section A.4, we used the predictions generated by our models on the temporal splits. In each specifications, we randomly shuffled the data to ensure that the mini-batch gradients are unbiased. If gradients are biased, training may not converge and accuracy may be lost.

B Machine Learning Models

B.1 Deep Neural Network (DNN)

Figure 12 illustrates an example of a two layer neural network. This neural network has 3 input units (denoted x_1, x_2, x_3), 4 hidden units, and 1 output unit. Let n_l denote the number of layers in this network ($n_l = 2$). We label layer l as L_l, where layer L_0 is the input layer, and layer $L_{L=2}$ is the output layer. The layers between the input ($l = 0$) and the output layer ($l = L$) are called hidden layers. Given this notation, there are $L - 1$ hidden layers, 1 in this specific example. A neural network without any hidden layers ($L = 1$) is a logistic regression model.

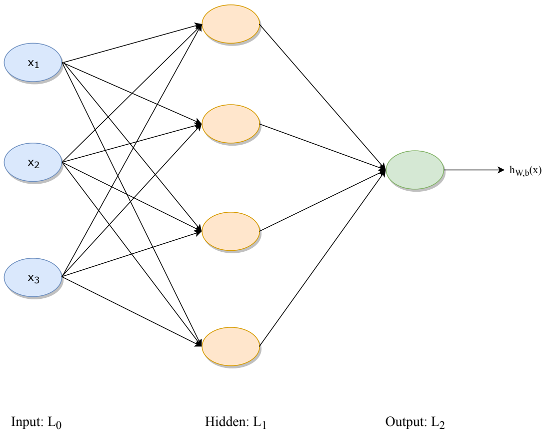


Figure 12: Two Layer Neural Network Example

There are two ways to increase the complexity a neural network: (1) increase the number of hidden layers and (2) increase the number of units in a given layer. Lower tier layers in the neural network learn simpler patterns, from which higher tier layers learn to produce more complex patterns. Given a sufficient number of neurons, neural networks can approximate continuous functions on compact sets arbitrarily well (see Hornik, Stinchcombe, and White (1989) and Hornik (1991)). This includes approximating interactions (i.e., the product and division of features). There are two main advantages of adding more layers over increasing the number of units to existing layers; (1) later layers build on early layers to learn features of greater complexity and (2) deep neural networks– those with three or more hidden layers–

need exponentially fewer neurons than shallow networks (Bengio et al. (2007) and Montúfar et al. (2014)).

In the neural network represented in Figure 12, the parameters to be estimated are $(W, b) = (W^{(0)}, b^{(0)}, W^{(1)}, b^{(1)})$, where $W_{ij}^{(l)}$ denotes the weight associated with the connection between unit j in layer l and unit i in layer $l + 1$, and $b_i^{(l)}$ is the bias associated with unit i in layer $l + 1$. Thus, in this example $W^{(0)} \in \mathbb{R}^{3\times4}, b^{(0)} \in \mathbb{R}^{4\times1}$ and $W^{(1)} \in \mathbb{R}^{1\times4}, b^{(1)} \in \mathbb{R}$. This implies that there are a total of 21 = (3+1)*4+5 parameters (four parameters to reach each neuron and five weights to aggregate the neurons into a single output). In general, the number of weight parameters in each hidden layer l is $N^{(l)}(1 + N^{(l-1)})$, plus $1 + N^{(L-1)}$ for the output layer, where $N^{(l)}$ denotes the number of neurons in each layer $l = 1,\ldots,\ L$.

Let $a_i^{(l)}$ denote the activation (e.g., output value) of unit i in layer l. Fix W and b, our neural network defines a hypothesis $h_{W,b}(x)$ that outputs a real number between 0 and 1.[23] Let $f(\cdot)$ denote the activation function that applies to vectors in an element-wise fashion. The computation this neural network represents, often referred to as forward propagation, can be written as:

$$z^{(1)} = W^{(0),T}x + b^{(0)}$$

$$a^{(1)} = f(z^{(1)})$$

$$z^{(2)} = W^{(1),T}a^{(1)} + b^{(1)}$$

$$h_{W,b}(x) = a^{(2)} = f(z^{(2)})$$

There are many choices to make when structuring a neural network, including the number of hidden layers, the number of neurons in each layer, and the activation functions. We built a number of network architectures having up to fifteen hidden layers.[24] All architectures are fully connected so each unit receives an input from all units in the previous layer.

Neural networks tend to be low-bias, high-variance models, which imparts them a tendency to over-fit the data. We apply dropout to each of the layers to avoid over-fitting (see Srivastava et al. (2014)). During training, neurons are randomly dropped (along with their connections) from the neural network with probability p (referred to as the dropout rate), which prevents complex co-adaptations on training data.

We apply the same activation function (rectified linear unit or RELU) at all nodes, which

[23] This is a property of the sigmoid activation function.

[24] The number of layers and the number of neurons in each layer, along with other hyperparameters of the model, are chosen by Tree-structured Parzen Estimator (TPE) approach. See Appendix C for more details.

is obtained via hyperparameter optimization,[25] and defined as:

$$\mathrm{RELU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

Let $N^{(l)}$ denote the number of neurons in each layer $l = 1, \ldots, L$. Define the output of neuron k in layer l as $z_k^{(l)}$. Then, define the vector of outputs (including the bias term $z_0^{(l)}$) for this layer as $z^{(l)} = (z_0^{(l)}, z_1^{(l)}, \ldots, z_{N^{(l)}}^{(l)})'$. For the input layer, define $z^{(0)} = (x_0^{(l)}, x_1^{(l)}, \ldots, x_{N^{(l)}}^{(l)})'$. Formally, the recursive output of the $l - th$ layer of the neural network is:

$$z^{(l)} = \mathrm{RELU}(W^{(l-1),T} z^{(l-1)} + b^{(l-1)}), \tag{18}$$

with final output:

$$h_\theta(x) = g(W^{(L-1),T} z^{(L-1)} + b^{(L-1)}). \tag{19}$$

The parameter specifying the neural network is:

$$\theta = (W_0, b_0, \ldots, W_{L-1}, b_{L-1}) \tag{20}$$

B.2 Decision Tree Models

The second component of our model is Extreme Gradient Boosting, which builds on decision tree models. Tree-based models split the data several times based on certain cutoff values in the explanatory variables.[26] A number of such models have become quite prevalent in the literature, most notably random forests (see Breiman (2001) and Butaru et al. (2016)) and Classification and Regression Trees, known as CART. We briefly review CART and then explain gradient boosting.

B.2.1 CART

There are a number of different decision tree-based algorithms. As an illustration of the approach, we describe Classification and Regression Trees or CART. CART models an outcome y_i for an instance i as follows:

$$\hat{y_i} = \hat{f}(x_i) = \sum_{m=1}^{M} c_m I\{x_i \in R_m\}, \tag{21}$$

[25] There are many potential choices for the nonlinear activation function, including the sigmoid, relu, and tanh.

[26] Splitting means that different subsets of the dataset are created, where each observation belongs to one subset. For a review on decision trees, see Khandani, Kim, and Lo (2010).

where each observation x_i belongs to exactly one subset R_m. The identity function I returns 1 if x_i is in R_m and 0 otherwise. If x_i falls into R_l, the predicted outcome is $\hat{y} = c_l$, where c_l is the mean of all training observations in R_l.

The estimation procedure takes a feature and computes the cut-off point that minimizes the Gini index of the class distribution of $\mathbf{y}$, which makes the two resulting subsets as different as possible. Once this is done for each feature, the algorithm uses the best feature to split the data into two subsets. The algorithm is then repeated until a stopping criterium is reached.

Tree-based models have a number of advantages that make them popular in applications. They are invariant to monotonic feature transformations and can handle categorical and continuous data in the same model. Like deep neural networks, they are well suited to capturing interactions between variables in the data. Specifically, a tree of depth L can capture $(L-1)$ interactions. The interpretation is straightforward, and provides immediate counterfactuals: "If feature x_j had been bigger / smaller than the split point, the prediction would have been $\bar{y}_0$ instead of $\bar{y}_1$." However, these models also have a number of limitations. They are poor at handling linear relationships, since tree algorithms rely on splitting the data using step functions, an intrinsically non-linear transformation. Trees also tend to be unstable, so that small changes in the training dataset might generate a different tree. They are also prone to overfitting to the training data. For more information on tree-based models see Molnar (2019).

B.2.2 Gradient Boosted Trees (GBT)

At each step m, $1 \leq m \leq M$, of gradient boosting, an estimator, h_m, is computed on the residuals from the previous models predictions. A critical part of gradient boosting method is regularization by shrinkage as proposed by Friedman (2001). This consists in modifying the update rule as follows:

$$F_m(x) = F_{m-1}(x) + \nu \gamma_m h_m(x), \tag{22}$$

where $h_m(x)$ represents a weak learner of fixed depth, γ_m is the step length and ν is the learning rate or shrinkage factor.

XGBoost is a fast implementation of Gradient Boosting, which has the advantages of fast speed and high accuracy. For classification, XGBoost combines the principles of decision trees and logistic regression, so that the output of our XGBoost model is a number between 0 and

1. For the remainder of the paper we refer to XGBoost as GBT.[27]

C Model Estimation

Our estimation consists of seven steps. First, we specify the loss function. Second, we choose the optimization algorithm. Third, we optimize the hyperparameters of our GBT model. Fourth, we train the GBT model. Fifth, we restrict our feature set using the GBT model. Sixth, we optimize the hyperparameters (including the weighting parameter for our hybrid models), and seventh, we train our models.

C.1 Loss Function

Suppose $\mathbf{y}$ is the ground truth vector of default, and $\hat{\mathbf{y}}$ is the estimate obtained directly from the last layer given input vector $\mathbf{x} = (x_1, x_2, \ldots, x_k)$. By construction, $y_i = \{0, 1\}$ and $\hat{y_i} \in [0, 1]$. We minimize the categorical cross-entropy loss function[28] to estimate the parameter specified in (7). We do this by choosing θ that minimizes the distance between the predicted $\hat{\mathbf{y}}$ and the actual $\mathbf{y}$ values. Given N training examples, the categorical cross-entropy loss can be written as:

$$L(\hat{y}, y) = -\frac{1}{N}\sum_{i=1}^{N}(y_i \cdot log(\hat{y_i}) + (1 - y_i) \cdot log(1 - \hat{y_i}) \tag{23}$$

We apply an iterative optimization algorithm to find the minimum of the categorical cross-entropy loss function. We next describe this algorithm.

C.2 DNN Optimization Algorithm

Deep learning models are computationally demanding due to their high degree of non-linearity, non-convexity and rich parameterization. Given the size of the data, gradient descent is impractical. We follow the standard approach of using stochastic gradient descent (SGD) to train our deep learning models (see Goodfellow et al. (2016)). Stochastic gradient descent is an iterative algorithm that uses small random subsets of the data to calculate the gradient of the objective function. Specifically, a subset of the data, referred to as a mini-batch (the size of the mini-batch is called the batch size), is loaded into memory and

[27] For more on XGBoost, see Chen, Lundberg, and Lee (2018) and Ren et al. (2017).

[28] Loss function measures the inconsistency between the predicted and the actual value. The performance of a model increases as the loss function decreases. There are several other types of loss functions, including mean squared error, hinge, and Poisson. The categorical cross-entropy is often used for classification problems.

the gradient is computed on this subset. The gradient is then updated, and the process is repeated until convergence.

We adopt the Adaptive Moment Estimation (Adam), a computationally efficient variant of the SGD introduced by (see Kingma and Ba (2014)) to train our neural networks. The Adam optimization algorithm can be summarized as follows:

1. Fix the learning rate α, the exponential decay rates for the moment estimates: β_1,$\beta_2 \in [0, 1)$, and the objective function. Initialize the parameter vector θ_0, the first and second moment vector m_0 and v_0 respectively, and the timestep t.

2. While θ_t does not converge, do the following:

   (a) Compute the gradients with respect to the objective function at timestep t:

   $$g_t = \nabla_\theta f_t(\theta_{t-1}) \tag{24}$$

   (b) Update the first and second moment estimates:

   $$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t \tag{25}$$

   $$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2 \tag{26}$$

   (c) Compute the bias-corrected first and second moment estimates:

   $$\hat{m_t} = \frac{m_t}{1 - \beta_1^t} \tag{27}$$

   $$\hat{v_t} = \frac{v_t}{1 - \beta_2^t} \tag{28}$$

   (d) Update the parameters:

   $$\theta_t = \theta_{t-1} - \frac{\alpha \cdot \hat{m}_t}{\sqrt{\hat{v_t}} + \epsilon} \tag{29}$$

The hyperparameters have intuitive interpretations and typically require little tuning. We apply the default setting suggested by the authors of Kingma and Ba (2014), these are $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-7}$.

C.3 GBT Algorithm

Fit a shallow tree (e.g., with depth L = 1). Using the prediction residuals from the first tree, fit a second tree with the same shallow depth L. Weight the predictions of the second tree by $\nu \in (0, 1)$ to prevent the model from overfitting the residuals, and then aggregate the forecasts of these two trees. Until a total of K trees is reached in the ensemble, at each step k, fit a shallow tree to the residuals from the model with k-1 trees, and add its prediction to the forecast of the ensemble with a shrinkage weight of ν.

C.4 Regularization

Neural networks are low-bias, high-variance models (i.e., they tend to overfit to their training data). We implement three routines to mitigate this. First, we apply dropout to each of the layers (see Srivastava et al. (2014)). During training, neurons are randomly dropped (along with their connections) from the neural network with probability p (referred to as the dropout rate), which prevents complex co-adaptations on training data.

Second, we implement "early stopping", a general machine learning regularization tool. After each time the optimization algorithm passes through the training data (i.e., referred to as an epoch), the parameters are gradually updated to minimize the prediction errors in the training data, and predictions are generated for the validation sample. We terminate the optimization when the validation sample loss has not decreased in the past 50 epochs. Early stopping is a popular substitute to l2 regularization, since it achieves regularization at a substantially lower computational cost.

Last, we use batch normalization (see Ioffe and Szegedy (2015)), a technique for controlling the variability of features across different regions of the network and across different datasets. It is motivated by the internal covariate shift, a phenomenon in which inputs of hidden layers follow different distributions than their counterparts in the validation sample. This problem is frequently encountered when fitting deep neural networks that involve many parameters and rather complex structures. For each hidden unit in each training step, the algorithm cross-sectionally de-means and variance standardizes the batch inputs to restore the representation power of the unit.

C.5 Feature Selection

First we remove a subset of features which are classified at the discretion of the lender and may be inconsistent across lenders, trades, and borrowers. These includes any variables pertaining to charge-off, derogatory in isolation, Fannie Mae, Freddie Mac, presence of outstanding govt agency debt, utility trades, and judgements. Then, we exclude features having limited impact on the model. To do so, we use data from 2004Q1 with test set from 2006Q1, and train a GBT model. We extract each features' feature importance, and sort our features based on this metric. We then iteratively remove features whose feature importance score is the lowest among our features, and train a GBT model with the corresponding feature set. We do this until there is only one feature left. We keep the lowest possible number of features that have the same or better predictive power than the model with the full set of features.

We then compute pairwise correlations for our full set of features, and add variables whose pairwise correlation exceeds 0.7 (since high correlation impacts feature importance). Next, we remove features so that no variables have a pairwise correlation over 0.9[29]. Our feature selection leaves us with a total of 88 features, and we report summary statistics for the forty most influential features (we rank their influence based on the perturbation exercise) in Table 10.

C.6 Hyperparameter Selection

Deep learning models require a number of hyperparameters to be selected. We follow the standard approach by cross-validating the hyperparameters via a validation set. We fix a training and validation set, and then train neural networks with different hyperparameters on the training set and compare the loss function on the validation set. We cross-validate the number of layers, the number of units per layer, the dropout rate, the batch size, and the activation function (i.e., the type of non-linearity) via Tree-structured Parzen Estimator (TPE) approach (see Bergstra et al. (2011)),[30] and select the hyperparameters with the lowest validation loss.

The training set for our out-of-sample hyperparameter optimization comes from 2004Q1, while the test set is from 2006Q1. Table 20 summarizes our machine learning model hyperparameters. For our neural network, we used 5 hidden layers, with 128-256-256-256-512 neurons per layer, RELU activation function, a batch size of 4096, a learning rate of 0.003, and a dropout rate of 50%. For our GBT, We found that a learning rate of 0.05, a max tree depth of 5, a max bin size of 256, with 1000 trees gave us good performance. All GBT models were run until their validation accuracy was non-improving for a hundred rounds and were trained on CPUs.

For the pooled sample prediction, we increased the batch size to 16,384 and the number of neurons per layers to 512,1024,2048,1024,512; while decreased the dropout rate to 20%, keeping the activation function, and the learning rate unchanged. We instituted early stopping with a patience of 1,000 for GBT, and trained a model of depth 6 with up to 10,000

[29] When it comes to tie-breakers, we keep the more generic feature; e.g., total debt balances are kept over total mortgage balances.

[30] We use TPE since it outperformed random search (see Bergstra et al. (2011)), which was shown to be both theoretically and empirically more efficient than standard techniques such as trials on a grid. Other widely used strategies are grid search and manual search.

Table 20: Hyperparameters for Machine Learning Models: Out-of-sample Exercise

Model	Tree Depth	# of Trees
CART	7	
RF	20	800
GBT	5	1000

trees and a learning rate of 0.3. We report the results of the best performing GBT.

C.7 Weighting

We use a grid search on our model trained on 2004Q1 data along with the 2006Q1 test data to find the optimal weights for our out-of-sample exercise, and we keep this weight constant going forward. For our pooled sample, we used the test data for finding the optimal weight[31].The results are reported in Table 21, and notice that the sample corresponds to Table 23. Based on this exercise, the optimal weight on the DNN is 0.2 for the out-of-sample exercise, and 0.7 for the pooled exercise.

Table 21: Weighting Schemes and Loss

Weight on DNN	Out-of-Sample Loss	Pooled Loss
0.2	0.3230	0.2778
0.3	0.3230	0.2745
0.1	0.3231	0.2823
0.4	0.3232	0.2721
0	0.3236	0.2890
0.5	0.3237	0.2705
0.6	0.3244	0.2695
0.7	0.3252	0.2693
0.8	0.3263	0.2700
0.9	0.3277	0.2717

Notes: Performance comparison of our hybrid DNN-GBT model under different weighting schemes. The results of predicted probabilities versus actual outcomes over the following 8Q (testing period) are used to calculate the loss metric for 90+ days delinquencies within 8Q. DNN refers to deep neural network, Source: Authors' calculations based on Experian Data.

[31] We only use the pooled sample for some interpretability and as a benchmark for our out-of-sample exercises, and such we are interested in finding the model with the best predictive power. We do not compare the pooled model's performance with the credit scores.

C.8 Implementation

We include our listed features for each individual. Since we work with panel data, there is a sample for each quarter of data. We train roughly 20 million samples, which takes up around 20 gigabytes of data. Our deep learning models are made up of millions of free parameters. Since the estimation procedure relies on computing gradients via backpropagation, which tends to be time and memory intensive, using conventional computing resources (e.g., desktop) would be impractical (if not infeasible). We address the time and memory intensity with two methods. First, to save memory, we use single precision floating point operations, which halves the memory requirements and results in a substantial computational speedup. Second, to accelerate the learning, we parallelized our computations and trained all of our models on a GPU cluster[32]. In our setting, GPU computations were over 40X faster than CPU for our deep neural networks. For a discussion on the impact of GPUs in deep learning see Schmidhuber (2015).

We conduct our analysis using Python 3.6.3 (Python Software Foundation), building on the packages numpy (Walt, Colbert, and Varoquaux (2011)), pandas (McKinney et al. (2010)) and matplotlib (Hunter (2007)). We develop our deep neural networks with keras (Chollet et al. (2015)) running on top of Google TensorFlow, a powerful library for large-scale machine learning on heterogenous systems (Abadi et al. (2016)). We run our machine learning algorithms using sci-kit learn (Pedregosa et al. (2011)) and (Chen and Guestrin (2016)).

D Model Comparisons

To better assess the validity of our approach, we compare our deep learning model to logistic regression and a number of other machine learning models. Deep learning models feature multiple hidden layers, designed to capture multi-dimensional feature interactions. By contrast, logistic regression can be interpreted as a neural network without any hidden layers.

D.1 Hidden Layers

To motivate our choice of deep learning, leading to our hybrid DNN-GBT model, we begin by illustrating the importance of hidden layers that enable us to capture multi-level interactions between features by comparing how neural networks of different depth perform on the pooled

[32] 1 node with 4 NVIDIA GeForceGTX1080 GPUs. The pooled model trains within 24 hours.

sample. For this exercise, we fix the number of neurons per layer at 512, and build neural networks up to 5 hidden layers.[33]

We benchmark our results against the logistic regression, which is a commonly used technique in credit scoring and can be interpreted as a neural network with no hidden layers. Table 22 reports the in- and out-of-sample behavior for neural networks with 0-5 hidden layers. The number of hidden layers measure the complexity of the network, and we found that the marginal improvements in performance beyond 3 layers are small. Table 22 also shows that applying dropout improves the out-of-sample fit for networks of higher depths. This demonstrates that dropout serves as an effective regularization tool and addresses overfitting for networks of greater depths.

Table 22: Neural Networks Comparison: Loss & Accuracy

Model	In-sample Loss		Out-of-sample Loss	
	w/o Dropout	Dropout	w/o Dropout	Dropout
Logistic Regression	0.3454	0.3454	0.3452	0.3452
1 layer	0.3167	0.3182	0.3179	0.3187
2 layers	0.3079	0.3139	0.3163	0.3160
3 layers	0.3018	0.3057	0.3149	0.3117
4 layers	0.2978	0.3017	0.3140	0.3101
5 layers	0.2955	0.3038	0.3138	0.3108

Model	In-sample Accuracy		Out-of-sample Accuracy	
	w/o Dropout	Dropout	w/o Dropout	Dropout
Logistic Regression	0.8564	0.8564	0.8564	0.8564
1 layer	0.8666	0.8660	0.8661	0.8658
2 layers	0.8707	0.8679	0.8670	0.8669
3 layers	0.8737	0.8718	0.8680	0.8689
4 layers	0.8756	0.8733	0.8685	0.8695
5 layers	0.8767	0.8726	0.8688	0.8691

Notes: In-sample and out-of-sample loss (categorical cross-entropy) and accuracy for neural networks of different depth and for logistic regression. Models are calibrated and evaluated on the pooled sample (2004Q1 - 2013Q4). Source: Authors' calculations based on Experian Data.

The results in Table 22 suggest that there are complex non-linear relationships among the features used as inputs in the model. This is further supported by the fact that permitting non-linear relationships between default behavior and explanatory variables produces the largest model improvement. Going from a linear model (0 layers) to the simplest non-linear model (1 layer) generates the most sizable reduction in out-of-sample loss. To see this from another angle, we plot the ROC curves for our neural networks considered in Figure 13.

[33] The architecture reported in Table 22 was not optimized. We picked 512, as it is exactly the number of neurons in the first layer for our pooled model.

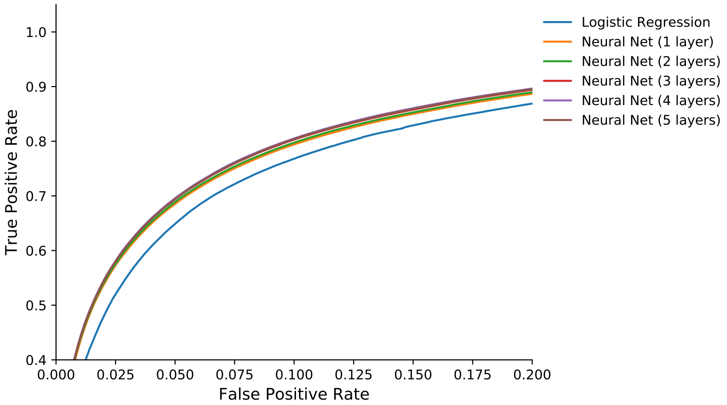


Figure 13: Out-of-Sample ROC Curves for Various Models with Dropout

Notes: Models are calibrated and evaluated on the pooled sample (2004Q1 - 2013Q4). Source: Authors' calculations based on Experian Data.

We can see that the logistic regression is dominated by all models that allow for non-linear relationships, while the improvements for deeper models are marginal.

D.2 Alternative Models

We next analyze a number of machine learning techniques that are possible alternatives to our hybrid model. These algorithms have been used in other credit scoring applications, and include decision trees (CART, see Khandani, Kim, and Lo (2010)), random forests (RF, see Butaru et al. (2016)), neural networks (see West (2000)), gradient boosting (GBT, see Xia et al. (2017)) and logistic regression. We use the out of sample loss as our main comparison metric, with lower loss values corresponding to better model performance. We tune the hyper-parameters for each model and present the results in Table 23 for our baseline 1 quarter training/validation samples. Our hybrid model performs the best, with gradient boosting coming second.

It is important to emphasize that these results do not imply that there does not exist a random forest or CART model that cannot outperform our hybrid model. The best model will depend on the specific sample. The exercise is intended to illustrate that the complexity of the model is proportional to its accuracy to a certain degree, and that deep neural networks improve substantially on shallow models, such as logistic regression.

Empirically, ensembles perform better when there is a significant diversity among the models (see Kuncheva and Whitaker (2003)). Table 24 shows the SHAP values for our hybrid DNN-GBT model in comparison to GBT and DNN models for the pooled sample. The results suggest there are significant differences between the DNN and GBT. For instance,

Table 23: Model Comparison: Out-of-Sample Loss

Training Window	Testing Window	Combined	GBT	RF	DNN	CART	Logistic
2004Q1	2006Q1	0.3230	0.3236	0.3266	0.3294	0.3416	0.3486
2004Q2	2006Q2	0.3187	0.3189	0.3224	0.3276	0.3372	0.3469
2004Q3	2006Q3	0.3165	0.3167	0.3206	0.3258	0.3349	0.3463
2004Q4	2006Q4	0.3200	0.3203	0.3240	0.3278	0.3396	0.3475
2005Q1	2007Q1	0.3208	0.3212	0.3253	0.3281	0.3426	0.3497
2005Q2	2007Q2	0.3217	0.3217	0.3261	0.3302	0.3424	0.3514
2005Q3	2007Q3	0.3247	0.3248	0.3292	0.3326	0.3454	0.3551
2005Q4	2007Q4	0.3278	0.3283	0.3323	0.3339	0.3490	0.3576
2006Q1	2008Q1	0.3265	0.3269	0.3312	0.3339	0.3476	0.3569
2006Q2	2008Q2	0.3275	0.3280	0.3317	0.3338	0.3473	0.3587
2006Q3	2008Q3	0.3259	0.3262	0.3298	0.3339	0.3453	0.3563
2006Q4	2008Q4	0.3259	0.3264	0.3291	0.3331	0.3457	0.3552
2007Q1	2009Q1	0.3210	0.3219	0.3244	0.3289	0.3407	0.3513
2007Q2	2009Q2	0.3198	0.3204	0.3238	0.3279	0.3397	0.3501
2007Q3	2009Q3	0.3177	0.3179	0.3211	0.3262	0.3386	0.3481
2007Q4	2009Q4	0.3143	0.3145	0.3180	0.3232	0.3352	0.3442
2008Q1	2010Q1	0.3153	0.3157	0.3190	0.3248	0.3347	0.3461
2008Q2	2010Q2	0.3159	0.3161	0.3197	0.3253	0.3366	0.3486
2008Q3	2010Q3	0.3161	0.3164	0.3195	0.3254	0.3356	0.3493
2008Q4	2010Q4	0.3171	0.3177	0.3208	0.3254	0.3365	0.3466
2009Q1	2011Q1	0.3154	0.3154	0.3181	0.3257	0.3323	0.3459
2009Q2	2011Q2	0.3179	0.3182	0.3209	0.3266	0.3350	0.3470
2009Q3	2011Q3	0.3168	0.3170	0.3191	0.3263	0.3347	0.3455
2009Q4	2011Q4	0.3154	0.3158	0.3182	0.3242	0.3316	0.3440
2010Q1	2012Q1	0.3142	0.3143	0.3172	0.3239	0.3312	0.3427
2010Q2	2012Q2	0.3162	0.3164	0.3195	0.3255	0.3335	0.3441
2010Q3	2012Q3	0.3151	0.3152	0.3186	0.3248	0.3321	0.3440
2010Q4	2012Q4	0.3167	0.3167	0.3204	0.3258	0.3353	0.3446
2011Q1	2013Q1	0.3141	0.3143	0.3172	0.3236	0.3324	0.3424
2011Q2	2013Q2	0.3139	0.3140	0.3174	0.3231	0.3322	0.3415
2011Q3	2013Q3	0.3123	0.3123	0.3159	0.3219	0.3301	0.3403
2011Q4	2013Q4	0.3104	0.3104	0.3143	0.3207	0.3274	0.3393
Average		0.3186	0.3189	0.3222	0.3272	0.3376	0.3480

Notes: Performance comparison of machine learning classification models of consumer default risk. The model calibrations are specified by the training and testing windows. The results of predicted probabilities versus actual outcomes over the following 8Q testing period are used to calculate the loss metric for 90+ days delinquencies within 8Q. Combined refers to the hybrid DNN-GBT model, DNN refers to deep neural network, RF refers to random forest, GBT refers to gradient boosted trees, while CART refers to decision tree. Source: Authors' calculations based on Experian Data.

monthly payment on mortgage trades is the third most important feature for GBT, while only twenty-fifth for DNN. Even more striking perhaps is that the number of credit cards is ranked most important for DNN, while only tenth for GBT.

Table 24: Shap Values across Models

Feature	Hybrid	Logistic	DNN	GBT
Worst status on any trades*	0.048 (1)	0.051 (2)	0.031 (3)	0.089 (2)
Months since the most recent 90+ days delinquency	0.037 (2)	0.048 (3)	0.015 (8)	0.091 (1)
Total debt balances*	0.031 (3)	0.028 (4)	0.023 (5)	0.062 (5)
Balance on collections*	0.031 (4)	0.002 (27)	0.017 (7)	0.065 (4)
Months since the oldest trade was opened	0.029 (5)	0.002 (30)	0.025 (4)	0.047 (8)
Number of credit cards*	0.028 (6)	0.027 (5)	0.037 (1)	0.033 (10)
Number of collections*	0.025 (7)	0.059 (1)	0.037 (2)	0.008 (30)
Balance on trades presently 90+ days delinquent	0.021 (8)	0.002 (29)	0.022 (6)	0.018 (19)
Months since the most recent 30-180days delinquency*	0.021 (9)	0.025 (6)	0.011 (11)	0.047 (7)
Monthly payment on mortgage debt*	0.017 (10)	0.0 (49)	0.005 (25)	0.066 (3)

Notes: This table reports the Shapley values for four selected machine learning classification models of consumer default risk. We sorted the features based on the feature's relative rank (in parentheses) using the hybrid model. Source: Authors' calculations based on Experian Data.

To see this from another angle, we grouped features into debt categories, and computed the SHAP values for each groups. The results are presented in Table 25, and once again shows significant differences between our models. For instance, features pertaining to collections contribute twice as much to our DNN model's prediction than to our GBT model's prediction. The ensemble approach can thus be thought of as providing diversification, which can reduce the variance of any of the two models. If one of the models puts too high of a weight on a feature, the other model may mitigate this effect.

Table 25: Shap Values across Debt Categories

			Model	
	Hybrid	Logistic	DNN	GBT
Feature Group				
Total Debt	0.4896	0.6241	0.4791	0.4607
Revolving Debt	0.2228	0.1816	0.2348	0.2374
Installment Debt	0.1895	0.1030	0.1594	0.2364
Collections	0.0981	0.0914	0.1266	0.0655

Notes: This table reports the aggregate absolute Shapley values for four selected machine learning classification models of consumer default risk. We grouped our features into debt categories, and computed the sum of the absolute SHAP values. Installment debt includes auto loans, mortgage debt, and student debt; while revolving debt includes credit and bankcard debt, and home equity line of credit trades. For ease of interpretability, we normalized our feature groups to 1 for each of our models. Source: Authors' calculations based on Experian Data.

Overall, the results summarized in this section suggest that deep learning is necessary to capture the complexity associated with default behavior, since all deep models perform substantially better than logistic regression. The importance of feature interaction reflects the complexity associated with default behavior. Additionally, our optimized model combines a deep neural network and gradient boosting and outperforms other machine learning models, such as random forests and decision trees, as well as deep neural networks and gradient boosting in isolation. However, all approaches show much stronger performance than logistic regression, suggesting that the main advantage is the adoption of a deep framework.

D.3 Expanded Training Data

We next compare the performance our our DNN to GBT by keeping our models' architecture the same, but expanding the training data. Table 26 looks at performance differences when we allow only the most recent 4 quarters for training, while Table 27 uses observations up till the date specified by the training window. These two exercises illustrate that while the performance of GBT remains similar, DNN benefits from having more data to train on.

D.4 Monotonicity Constraints

We next look at the performance trade-offs in placing monotonicity constraints on some of our features for our GBT model. Placing constraints on certain features might be required by legislation and can be desirable from a fairness standpoint (e.g., individuals who are late on their debt should have higher default risk). Monotonicity constraints are easily implementable and enforceable for GBT: one just needs to specify the set features to constrain and the corresponding relationship between the feature and the model's output (i.e., increasing or decreasing).[34]

We investigated constraining two sets of features. Under Regime I (R I), we placed constraints on each of our "Worst status" features. In particular, under these constraints, we require that our GBT model's output be increasing in these features. To illustrate this, suppose we have to individuals whose credit report file only differs in one of the "Worst status" features. Then, under Regime I, the individual whose "Worst status" feature is higher will have a higher predicted probability of default.[35] Imposing Regime I only marginally affects the model's performance, and results in a slightly lower average loss. We conjecture that our unconstrained model implicitly learns this relationship, but slightly overfits to the training data, which results in marginally worse model performance. This is also in line

[34] As of now, imposing monotonicity constraints are not possible for standard DNN models.You et al. (2017) proposes deep lattice networks that are monotonic with respect to a user-specified set of inputs as an alternative to standard DNNs, while Gupta et al. (2020) implements a special loss function to achieve partial monotonicity for DNNs.

[35] "Worst status" features range from current → unrated → 30 days late → 60 days late → 90 days late → 120-180 days late → derogatory, with current having the lowest value.

with our pooled sample SHAP value results, where ”Worst status” features were positively associated with default risk.

Under Regime II, we place negative constraints on our credit limit variables, length of credit and temporal proximity to delinquency variables, while positive constraints on features pertaining to ”Worst status”, number of delinquencies, bankruptcies, collections, utilization and fraction of delinquent balances. Notice, however that enforcing Regime II results in a sizeable performance drop, which could be due to the non-linear relationships we illustrated in Section 3. This finding suggests that one must be careful in selecting features to constrain, as it might result in significant performance drops.

Table 26: Model Comparison: DNN vs. GBT, 1 Year

Training Window Start	Training Window End	Testing Window	AUC-score		Loss	
			DNN	GBT	DNN	GBT
2004Q1	2004Q1	2006Q1	0.9219	0.9239	0.3294	0.3236
2004Q1	2004Q2	2006Q2	0.9222	0.9249	0.3279	0.3193
2004Q1	2004Q3	2006Q3	0.9236	0.9262	0.3245	0.3165
2004Q1	2004Q4	2006Q4	0.9232	0.9252	0.3260	0.3198
2004Q2	2005Q1	2007Q1	0.9234	0.9257	0.3289	0.3208
2004Q3	2005Q2	2007Q2	0.9237	0.9256	0.3281	0.3222
2004Q4	2005Q3	2007Q3	0.9224	0.9250	0.3334	0.3247
2005Q1	2005Q4	2007Q4	0.9220	0.9240	0.3342	0.3276
2005Q2	2006Q1	2008Q1	0.9221	0.9246	0.3364	0.3276
2005Q3	2006Q2	2008Q2	0.9223	0.9240	0.3349	0.3291
2005Q4	2006Q3	2008Q3	0.9232	0.9250	0.3338	0.3269
2006Q1	2006Q4	2008Q4	0.9236	0.9254	0.3319	0.3266
2006Q2	2007Q1	2009Q1	0.9248	0.9272	0.3290	0.3224
2006Q3	2007Q2	2009Q2	0.9254	0.9274	0.3268	0.3214
2006Q4	2007Q3	2009Q3	0.9263	0.9285	0.3246	0.3178
2007Q1	2007Q4	2009Q4	0.9279	0.9301	0.3207	0.3138
2007Q2	2008Q1	2010Q1	0.9274	0.9299	0.3223	0.3150
2007Q3	2008Q2	2010Q2	0.9267	0.9297	0.3236	0.3149
2007Q4	2008Q3	2010Q3	0.9266	0.9294	0.3234	0.3149
2008Q1	2008Q4	2010Q4	0.9267	0.9291	0.3238	0.3165
2008Q2	2009Q1	2011Q1	0.9271	0.9299	0.3221	0.3142
2008Q3	2009Q2	2011Q2	0.9254	0.9283	0.3257	0.3172
2008Q4	2009Q3	2011Q3	0.9254	0.9282	0.3256	0.3168
2009Q1	2009Q4	2011Q4	0.9257	0.9285	0.3242	0.3159
2009Q2	2010Q1	2012Q1	0.9261	0.9292	0.3222	0.3137
2009Q3	2010Q2	2012Q2	0.9250	0.9278	0.3244	0.3160
2009Q4	2010Q3	2012Q3	0.9249	0.9277	0.3230	0.3146
2010Q1	2010Q4	2012Q4	0.9245	0.9273	0.3243	0.3158
2010Q2	2011Q1	2013Q1	0.9254	0.9280	0.3216	0.3134
2010Q3	2011Q2	2013Q2	0.9248	0.9277	0.3220	0.3132
2010Q4	2011Q3	2013Q3	0.9256	0.9283	0.3189	0.3114
2011Q1	2011Q4	2013Q4	0.9255	0.9284	0.3192	0.3103
Average			0.9247	0.9272	0.3262	0.3186

Notes: Performance comparison of the two best performing machine learning classification models of consumer default risk. The model calibrations are specified by the training and testing windows. The results of predicted probabilities versus actual outcomes over the following 8Q (testing period) are used to calculate the loss metric and the AUC-score for 90+ days delinquencies within 8Q. DNN refers to deep neural network, GBT refers to gradient boosted trees. Source: Authors' calculations based on Experian Data.

Table 27: Model Comparison: DNN vs. GBT, Full

Training Window*	Testing Window	AUC-score		Loss	
		DNN	GBT	DNN	GBT
2004Q1	2006Q1	0.9219	0.9239	0.3280	0.3236
2004Q2	2006Q2	0.9226	0.9249	0.3264	0.3193
2004Q3	2006Q3	0.9238	0.9262	0.3238	0.3165
2004Q4	2006Q4	0.9234	0.9252	0.3264	0.3198
2005Q1	2007Q1	0.9234	0.9257	0.3281	0.3212
2005Q2	2007Q2	0.9230	0.9255	0.3309	0.3228
2005Q3	2007Q3	0.9223	0.9249	0.3342	0.3255
2005Q4	2007Q4	0.9217	0.9238	0.3366	0.3288
2006Q1	2008Q1	0.9226	0.9245	0.3362	0.3292
2006Q2	2008Q2	0.9223	0.9240	0.3363	0.3308
2006Q3	2008Q3	0.9229	0.9250	0.3354	0.3285
2006Q4	2008Q4	0.9229	0.9252	0.3354	0.3284
2007Q1	2009Q1	0.9246	0.9271	0.3314	0.3237
2007Q2	2009Q2	0.9249	0.9273	0.3292	0.3225
2007Q3	2009Q3	0.9260	0.9284	0.3257	0.3188
2007Q4	2009Q4	0.9278	0.9300	0.3213	0.3146
2008Q1	2010Q1	0.9277	0.9300	0.3224	0.3155
2008Q2	2010Q2	0.9277	0.9300	0.3215	0.3147
2008Q3	2010Q3	0.9276	0.9299	0.3211	0.3140
2008Q4	2010Q4	0.9278	0.9297	0.3205	0.3147
2009Q1	2011Q1	0.9285	0.9308	0.3188	0.3119
2009Q2	2011Q2	0.9265	0.9291	0.3224	0.3152
2009Q3	2011Q3	0.9267	0.9292	0.3216	0.3142
2009Q4	2011Q4	0.9269	0.9291	0.3206	0.3141
2010Q1	2012Q1	0.9268	0.9296	0.3206	0.3125
2010Q2	2012Q2	0.9255	0.9282	0.3226	0.3151
2010Q3	2012Q3	0.9252	0.9278	0.3214	0.3140
2010Q4	2012Q4	0.9246	0.9274	0.3236	0.3150
2011Q1	2013Q1	0.9256	0.9281	0.3198	0.3131
2011Q2	2013Q2	0.9251	0.9277	0.3202	0.3130
2011Q3	2013Q3	0.9257	0.9282	0.3192	0.3112
2011Q4	2013Q4	0.9256	0.9283	0.3181	0.3101
Average		0.9250	0.9273	0.3256	0.3185

Notes: Performance comparison of the two best performing machine learning classification models of consumer default risk. The model calibrations are specified by the training and testing windows. * implies that all data was used up to the quarter specified. The results of predicted probabilities versus actual outcomes over the following 8Q (testing period) are used to calculate the loss metric and the AUC-score for 90+ days delinquencies within 8Q. DNN refers to deep neural network, GBT refers to gradient boosted trees. Source: Authors' calculations based on Experian Data.

Table 28: Constraining Model Behavior

Training Window	Testing Window	AUC-score			Loss		
		UN	R I	R II	UN	R I	R II
2004Q1	2006Q1	0.9239	0.9239	0.9222	0.3236	0.3237	0.3262
2004Q2	2006Q2	0.9247	0.9247	0.9230	0.3189	0.3189	0.3219
2004Q3	2006Q3	0.9257	0.9257	0.9239	0.3167	0.3167	0.3199
2004Q4	2006Q4	0.9246	0.9247	0.9229	0.3203	0.3202	0.3235
2005Q1	2007Q1	0.9254	0.9254	0.9237	0.3212	0.3212	0.3243
2005Q2	2007Q2	0.9255	0.9255	0.9238	0.3217	0.3217	0.3249
2005Q3	2007Q3	0.9249	0.9249	0.9231	0.3248	0.3248	0.3281
2005Q4	2007Q4	0.9235	0.9235	0.9218	0.3283	0.3284	0.3313
2006Q1	2008Q1	0.9245	0.9246	0.9229	0.3269	0.3267	0.3298
2006Q2	2008Q2	0.9242	0.9244	0.9227	0.3280	0.3277	0.3308
2006Q3	2008Q3	0.9253	0.9254	0.9237	0.3262	0.3258	0.3291
2006Q4	2008Q4	0.9255	0.9257	0.9240	0.3264	0.3260	0.3292
2007Q1	2009Q1	0.9274	0.9276	0.9259	0.3219	0.3213	0.3246
2007Q2	2009Q2	0.9277	0.9278	0.9261	0.3204	0.3201	0.3237
2007Q3	2009Q3	0.9284	0.9285	0.9269	0.3179	0.3178	0.3210
2007Q4	2009Q4	0.9297	0.9299	0.9284	0.3145	0.3142	0.3176
2008Q1	2010Q1	0.9296	0.9296	0.9281	0.3157	0.3157	0.3190
2008Q2	2010Q2	0.9293	0.9294	0.9277	0.3161	0.3161	0.3200
2008Q3	2010Q3	0.9289	0.9290	0.9273	0.3164	0.3165	0.3202
2008Q4	2010Q4	0.9286	0.9287	0.9269	0.3177	0.3178	0.3216
2009Q1	2011Q1	0.9293	0.9294	0.9278	0.3154	0.3155	0.3190
2009Q2	2011Q2	0.9279	0.9279	0.9261	0.3182	0.3184	0.3222
2009Q3	2011Q3	0.9281	0.9281	0.9264	0.3170	0.3170	0.3208
2009Q4	2011Q4	0.9285	0.9285	0.9267	0.3158	0.3157	0.3195
2010Q1	2012Q1	0.9290	0.9291	0.9273	0.3143	0.3144	0.3181
2010Q2	2012Q2	0.9277	0.9277	0.9260	0.3164	0.3165	0.3202
2010Q3	2012Q3	0.9275	0.9276	0.9257	0.3152	0.3153	0.3192
2010Q4	2012Q4	0.9269	0.9269	0.9250	0.3167	0.3168	0.3209
2011Q1	2013Q1	0.9278	0.9278	0.9257	0.3143	0.3143	0.3186
2011Q2	2013Q2	0.9274	0.9274	0.9255	0.3140	0.3141	0.3180
2011Q3	2013Q3	0.9280	0.9280	0.9260	0.3123	0.3124	0.3164
2011Q4	2013Q4	0.9283	0.9282	0.9262	0.3104	0.3106	0.3148
Average		0.9270	0.9270	0.9253	0.3189	0.3188	0.3223

Notes: Performance comparison of GBT models of consumer default risk under various monotonicity constraint regimes. UN denotes the unconstrained model, while R I and R II are the models under Regime I and Regime II respectively. The model calibrations are specified by the training and testing windows. The results of predicted probabilities versus actual outcomes over the following 8Q testing period are used to calculate the loss metric for 90+ days delinquencies within 8Q. Source: Authors' calculations based on Experian Data.

References

Abadi, Martín, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. "Tensorflow: a system for large-scale machine learning." *OSDI*, Volume 16. 265–283.

Agarwal, Sumit, Souphala Chomsisengphet, Neale Mahoney, and Johannes Stroebel. 2014. "Regulating consumer financial products: Evidence from credit cards." *The Quarterly Journal of Economics* 130 (1): 111–164.

———. 2015, September. "Do Banks Pass Through Credit Expansions to Consumers Who Want to Borrow?" Working paper 21567, National Bureau of Economic Research.

Athey, Susan, and Guido W Imbens. 2019. "Machine learning methods that economists should know about." *Annual Review of Economics*, vol. 11.

Athreya, Kartik, Xuan S Tam, and Eric R Young. 2012. "A quantitative theory of information and unsecured credit." *American Economic Journal: Macroeconomics* 4 (3): 153–83.

Ausubel, Lawrence M. 1991. "The failure of competition in the credit card market." *The American Economic Review*, pp. 50–81.

Barbaglia, Luca, Sebastiano Manzan, and Elisa Tosetti. 2020. "Forecasting Loan Default in Europe with Machine Learning." *Available at SSRN 3605449.*

Bengio, Yoshua, Yann LeCun, et al. 2007. "Scaling learning algorithms towards AI." *Large-scale kernel machines* 34 (5): 1–41.

Bergstra, James S, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. 2011. "Algorithms for hyper-parameter optimization." *Advances in neural information processing systems.* 2546–2554.

Branzoli, Nicola, and Ilaria Supino. 2020. "FinTech Credit: A Critical Review of Empirical Research Literature." *Bank of Italy Occasional Paper*, no. 549.

Breiman, Leo. 2001. "Random forests." *Machine learning* 45 (1): 5–32.

Butaru, Florentin, Qingqing Chen, Brian Clark, Sanmay Das, Andrew W Lo, and Akhtar Siddique. 2016. "Risk and risk management in the credit card industry." *Journal of Banking & Finance* 72:218–239.

Calem, Paul S, and Loretta J Mester. 1995. "Consumer behavior and the stickiness of credit-card interest rates." *The American Economic Review* 85 (5): 1327–1336.

Chatterjee, Satyajit, Dean Corbae, Makoto Nakajima, and Jose-Victor Rios-Rull. 2007. "A quantitative theory of unsecured consumer credit with risk of default." *Econometrica* 75 (6): 1525–1589.

Chatterjee, Satyajit, Dean Corbae, and Jose-Victor Rios-Rull. 2011. "A theory of credit scoring and competitive pricing of default risk." *Unpublished paper, University of Minnesota.[672]*, vol. 31.

Chen, Hugh, Scott Lundberg, and Su-In Lee. 2018. "Hybrid Gradient Boosting Trees and Neural Networks for Forecasting Operating Room Data." *arXiv preprint arXiv:1801.07384.*

Chen, Tianqi, and Carlos Guestrin. 2016. "XGBoost: A Scalable Tree Boosting System." *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16. New York, NY, USA: ACM, 785–794.

Chollet, François, et al. 2015. "Keras: Deep learning library for theano and tensorflow." *URL: https://keras. io/k* 7, no. 8.

Corbae, Dean, and Andrew Glover. 2018, September. "Employer Credit Checks: Poverty Traps versus Matching Efficiency." Working paper 25005, National Bureau of Economic Research.

Friedman, Jerome H. 2001. "Greedy function approximation: a gradient boosting machine." *Annals of statistics*, pp. 1189–1232.

Fuster, Andreas, Paul Goldsmith-Pinkham, Tarun Ramadorai, and Ansgar Walther. 2018. "Predictably unequal? the effects of machine learning on credit markets." *The Effects of Machine Learning on Credit Markets (November 6, 2018).*

Goodfellow, Ian, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. 2016. *Deep learning.* Volume 1. MIT press Cambridge.

Gupta, Akhil, Naman Shukla, LLC Deepair, Lavanya Marla, Arinbjörn Kolbeinsson, and Kartik Yellepeddi. 2020. "How to Incorporate Monotonicity in Deep Networks While Preserving Flexibility?"

Hornik, Kurt. 1991. "Approximation capabilities of multilayer feedforward networks." *Neural Networks* 4 (2): 251 – 257.

Hornik, Kurt, Maxwell Stinchcombe, and Halbert White. 1989. "Multilayer feedforward networks are universal approximators." *Neural Networks* 2 (5): 359 – 366.

Hunter, John D. 2007. "Matplotlib: A 2D graphics environment." *Computing in science & engineering* 9 (3): 90–95.

Ioffe, Sergey, and Christian Szegedy. 2015. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." *arXiv preprint arXiv:1502.03167.*

Khandani, Amir E, Adlar J Kim, and Andrew W Lo. 2010. "Consumer credit-risk models via machine-learning algorithms." *Journal of Banking & Finance* 34 (11): 2767–2787.

Kingma, Diederik P, and Jimmy Ba. 2014. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980.*

Kuncheva, Ludmila I, and Christopher J Whitaker. 2003. "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy." *Machine learning* 51 (2): 181–207.

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

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

Livshits, Igor, James MacGee, and Michele Tertilt. 2007. “Consumer bankruptcy: A fresh start.” *American Economic Review* 97 (1): 402–418.

Lundberg, Scott M, and Su-In Lee. 2017. “A Unified Approach to Interpreting Model Predictions.” In *Advances in Neural Information Processing Systems 30*, edited by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, 4765–4774. Curran Associates, Inc.

McKinney, Wes, et al. 2010. “Data structures for statistical computing in python.” *Proceedings of the 9th Python in Science Conference*, Volume 445. Austin, TX, 51–56.

Molnar, Christoph. 2019. *Interpretable Machine Learning.* `https://christophm.github.io/interpretable-ml-book/`.

Montúfar, Guido, Razvan Pascanu, Kyunghyun Cho, and Yoshua Bengio. 2014. “On the Number of Linear Regions of Deep Neural Networks.” *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS’14. Cambridge, MA, USA: MIT Press, 2924–2932.

Moscatelli, Mirko, Fabio Parlapiano, Simone Narizzano, and Gianluca Viggiano. 2020. “Corporate default forecasting with machine learning.” *Expert Systems with Applications*, p. 113567.

Mullainathan, Sendhil, and Jann Spiess. 2017. “Machine learning: an applied econometric approach.” *Journal of Economic Perspectives* 31 (2): 87–106.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. “Scikit-learn: Machine learning in Python.” *Journal of machine learning research* 12 (Oct): 2825–2830.

Ren, Xudie, Haonan Guo, Shenghong Li, Shilin Wang, and Jianhua Li. 2017. “A novel image classification method with CNN-XGBoost model.” *International Workshop on Digital Watermarking*. Springer, 378–390.

Schmidhuber, Jürgen. 2015. “Deep learning in neural networks: An overview.” *Neural networks* 61:85–117.

Sirignano, Justin, Apaar Sadhwani, and Kay Giesecke. 2018. “Deep Learning for Mortgage Risk.” *Available at SSRN.*

Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. “Dropout: a simple way to prevent neural networks from overfitting.” *The Journal of Machine Learning Research* 15 (1): 1929–1958.

Vamossy, Domonkos F. 2020. “Investor Emotions and Earnings Announcements.” *Available at SSRN 3626025.*

Walt, Stéfan van der, S Chris Colbert, and Gael Varoquaux. 2011. “The NumPy array: a structure for efficient numerical computation.” *Computing in Science & Engineering* 13 (2): 22–30.

West, David. 2000. “Neural network credit scoring models.” *Computers & Operations Research* 27 (11-12): 1131–1152.

Xia, Yufei, Chuanzhe Liu, YuYing Li, and Nana Liu. 2017. "A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring." *Expert Systems with Applications* 78:225–241.

You, Seungil, David Ding, Kevin Canini, Jan Pfeifer, and Maya Gupta. 2017. "Deep lattice networks and partial monotonic functions." *Advances in neural information processing systems.* 2981–2989.