Recession forecasting with big data *

Lauri Nevasalmi †

February 5, 2021

Abstract

In this paper, a large amount of different financial and macroeconomic variables are used to predict the U.S. recession periods. We propose a new cost-sensitive extension to the gradient boosting model which can take into account the class imbalance problem of the binary response variable. The class imbalance, caused by the scarcity of recession periods in our application, is a problem that is emphasized with high-dimensional datasets. Our empirical results show that the introduced cost-sensitive extension outperforms the traditional gradient boosting model in both in-sample and out-of-sample forecasting. Among the large set of candidate predictors, different types of interest rate spreads turn out to be the most important predictors when forecasting U.S. recession periods.

Keywords: Recession forecasting, business cycle, machine learning, gradient boosting, class imbalance
JEL classification: C22, C25, C53, C55, E32

* The author would like to thank Heikki Kauppi, Henri Nyberg, Simone Maxand, Charlotte Christiansen and seminar participants at the Annual Meeting of the Finnish Economic Association and FDPE Econometrics workshop for constructive comments. The financial support from the Emil Aaltonen Foundation and the Academy of Finland (grant 321968) is gratefully acknowledged.

† Department of Mathematics and Statistics, University of Turku. FI-20014 University of Turku, Finland. E-mail: lauri.nevasalmi@utu.fi

1 Introduction

Recessions are painful periods with a significant and widespread decline in economic activity. Early warning signals of recessions would be important for different kinds of economic agents. Households, firms, policymakers and central bankers could all utilize the information concerning upcoming economic activity in their decision making. The probability of a recession is fairly straightforward to interpret and can be easily taken into consideration in all kinds of economic decision making.

But what are the indicators that consistently lead recessions? Since the early work of Estrella and Mishkin (1998) there has been a large amount of empirical research concerning the predictive content of different economic and financial variables (see e.g., Nyberg, 2010; Liu and Moench, 2016). The amount of potential recession indicators is growing rapidly as the constraints related to data-availability and computational power keep diminishing. Traditionally used binary logit and probit models can only handle small predictor sets at a time, which makes the search for the best predictors quite difficult.

Recent developments in the machine learning literature provide a solution to this problem. State of the art supervised learning algorithm called gradient boosting is able to do variable selection and model estimation simultaneously. Non-parametric boosting can handle huge predictor sets and the estimated conditional probability function can take basically any kind of form. The main objective of this research is to explore how we can exploit high-dimensional datasets when making recession forecasts with the gradient boosting model.

The business cycle consists of positive and negative fluctuations around the long-run growth rate of the economy. These fluctuations are also known as expansions and recessions. The official business cycle chronology for the U.S. is published by the National Bureau of Economic Research (NBER). Recessions are shorter events compared to expansion periods leading to quite heavily imbalanced binary class labels. In our dataset less than 14 percent of the monthly observations are classified as recessions. This class imbalance and the effects on classification is well covered in the machine learning literature (see e.g., Galar et al., 2012). Suprisingly the scarcity of recession periods has not been properly taken into consideration in previous economic research.

Two approaches are usually considered when dealing with imbalanced classes: resampling techniques and cost-sensitive learning methods (see e.g., He and Garcia, 2009). Resampling is the easiest and most commonly used alternative. The dataset could be balanced by drawing a random sample without replacement from the majority class, which is called undersampling. In the recession forecasting setup the size of the dataset is already very limited so this could create problems when estimating the model, especially with high-dimensional data. In the oversampling approach the idea is to

sample with replacement from the minority class. He and Garcia (2009) argue that the duplicate observations from the minority class can lead to overfitting.

Instead of replicating existing observations from the minority class one could learn the characteristics in this class and create synthetic samples based on feature space similarities. This synthetic minority oversampling technique also known as SMOTE is a popular alternative when dealing with imbalanced data. Blagus and Lusa (2013) however find that variable selection is needed before running SMOTE on high-dimensional datasets.

Cost-sensitive learning methods can take the class imbalance into account without artificially manipulating the dataset. In a variety of real-life classification problems, such as recession forecasting or fraud detection, misclassifying the minority class can be considered very costly. The cost-sensitivity can be incorporated into the model by attaching a higher penalty for misclassifying the minority class. Several modified versions of the adaboost algorithm by Freund and Schapire (1996) exist, where the weight updating rule of the original algorithm is modified to better account for the class imbalance (see e.g., Sun et al., 2007; Fan et al., 1999; Ting, 2000).

This is natural since weight updating is a crucial part of the adaboost algorithm designed purely for classification problems. However this is not the case with the more general gradient boosting algorithm presented by Friedman (2001) that can handle variety of problems beyond classification and the cost-sensitivity have to be incorporated otherwise. We propose a cost-sensitive extension to the gradient boosting model by introducing a binary class weight to each observation in the dataset that reflect the asymmetric misclassification costs. To the best of our knowledge cost-sensitive gradient boosting model using class weights has not been utilized in previous economic research.

The traditional gradient boosting model has been utilized in previous economic research with mixed results. Ng (2014) uses the gradient boosting model with stump regression trees to predict recession periods in the U.S. The dataset used by Ng (2014) has a fairly large predictor set and is from the same source as the dataset used in this paper. With this model setup Ng (2014) concludes that the gradient boosting model is far from perfect in forecasting recessions.

Berge (2015) uses a smaller predictor set to forecast U.S. recessions with the gradient boosting model. The results show how boosting outperforms other model selection techniques such as Bayesian model averaging. Moreover, the results highlight the importance of non-linearity in recession forecasting as boosting with non-linear smoothing splines outperforms boosting with a linear final model. Döpke, Fritsche and Pierdzioch (2017) succesfully forecast German recession periods with the gradient boosting model using regression trees. Unlike Ng (2014) they build larger trees which allow for potential interaction terms between predictors. This approach is used in this study as well.

Our results confirm the finding of Blagus and Lusa (2017) who note that the per-

formance of a gradient boosting model can be rather poor with high class imbalance, especially when a high-dimensional dataset is used. The out-of-sample forecasting ability of the traditional gradient boosting model is quite heavily deteriorated compared to the in-sample results. The cost-sensitive extension to the gradient boosting model using class weights can take the class imbalance problem into account and produces strong warning signals for the U.S. recessions with different forecasting horizons.

The cost-sensitive gradient boosting models estimated using huge predictor sets rely heavily on different kinds of interest rate spreads. This is also the case with the short and medium term forecasting horizons although different variables related to the real economy are also available in the dataset. The internal model selection capability of gradient boosting confirms that predictors with predictive power beyond the term spread are quite hard to find (see e.g., Estrella and Mishkin, 1998; Liu and Moench, 2016).

The results also show how the chosen lag length for a predictor can vary substantially from the forecasting horizon considered. A similar observation has been made by Kauppi and Saikkonen (2008) in the conventional probit model. The term spread is the dominant predictor when forecasting recessions one year ahead, which is a common finding in the previous literature (see e.g., Dueker, 1997; Estrella and Mishkin, 1998).

The rest of the paper is organized as follows. The gradient boosting framework and the cost-sensitive extension to the gradient boosting model are introduced in Section 2. The dataset and the empirical analysis are presented in Section 3. Section 4 concludes.

2 Methodology

The following theoretical framework for the gradient boosting model follows closely the original work of Friedman (2001).

2.1 Gradient boosting

Considering two stochastic processes y_t and $\mathbf{x}_{t-k}$ of which y_t is a binary dependent variable of form

$$y_t = \begin{cases} 1, & \text{if economy in recession at time t} \\ 0, & \text{if economy in expansion at time t} \end{cases} \tag{1}$$

and $\mathbf{x}_{t-k}$ is a $p \times 1$ vector of predictive variables. The lag length k of each predictor must satisfy the condition $k \geq h$, where h is the forecasting horizon. If $E_{t-k}(\cdot)$ and $P_{t-k}(\cdot)$ denote conditional expectation and conditional probability given the information set available at time $t-k$ and by assuming the logistic transform $\Lambda(\cdot)$ the conditional

probability can be written as

$$E_{t-k}(y_t) = P_{t-k}(y_t = 1) = p_t = \Lambda(F(\mathbf{x}_{t-k})). \tag{2}$$

We can model this conditional probability by estimating the function $F(\mathbf{x}_{t-k})$ with the gradient boosting model. Exponential loss and binomial deviance are popular alternatives for the loss function to be minimized with binary classification problems. These are second order equivalent (Friedman, Hastie and Tibshirani, 2000). In this research the conditional probability is estimated with the gradient boosting model by minimizing the binomial deviance loss function.

In the general estimation problem the goal is to find the function $F(\mathbf{x}_{t-k})$ that minimizes the expected loss of some predefined loss function

$$\widehat{F}(\mathbf{x}_{t-k}) = \underset{F(\mathbf{x}_{t-k})}{\arg\min}\, E\left[\mathcal{L}\left(y_t, F(\mathbf{x}_{t-k})\right)\right]. \tag{3}$$

Even for a simple parametric model, where $F(\mathbf{x}_{t-k})$ is assumed to be a linear function of the covariates, numerical optimization techniques are usually needed for solving the parameter vector that minimizes the expected loss in equation (3). Steepest descent optimization technique is a simple alternative. The parameter search using steepest descent can be summarized with the following equation

$$\boldsymbol{\beta}^* = \sum_{m=0}^{M} \boldsymbol{\beta}_m = \sum_{m=0}^{M} -\delta_m \mathbf{g}_m, \tag{4}$$

where $\boldsymbol{\beta}_0$ is the initial guess and $\{\boldsymbol{\beta}_m\}_{m=1}^{M}$ are steps towards the optimal solution. The negative gradient vector $-\mathbf{g}_m$ determines the direction of each step and δ_m is the stepsize obtained by a line search.

With gradient boosting the optimization takes place in the function space instead of the conventional parameter space. Similarly as in the parametric case numerical optimization methods are needed when searching for the optimal function. Some further assumptions are required in order to make the numerical optimization in the function space feasible with finite datasets. By restricting the function search to some parameterized class of functions the solution to numerical optimization can be written as

$$F^*(\mathbf{x}_{t-k}) = \sum_{m=0}^{M} f_m(\mathbf{x}_{t-k}) = \sum_{m=0}^{M} \delta_m b(\mathbf{x}_{t-k}; \boldsymbol{\gamma}_m), \tag{5}$$

where δ_m is the stepsize obtained by line search as in equation (4). Now the step "direction" is given by the function $b(\mathbf{x}_{t-k}; \boldsymbol{\gamma}_m)$ also known as the base learner function. This can be a simple linear function or highly non-linear such as splines or regression trees. In this paper regression trees are used and the parameter vector $\boldsymbol{\gamma}_m$ consists of the

splitting variables and splitpoints of the regression tree. Equation (5) also incorporates the original idea of boosting. The possibly very complex final ensemble $F(\mathbf{x}_{t-k})$ with strong predictive ability is a sum of the fairly simple base learner functions $f_m(\mathbf{x}_{t-k})$.

Using the sample counterpart of the loss function in equation (3) and by plugging in the additive form introduced in equation (5) the estimation problem can be written as

$$\min_{\{\delta_m,\boldsymbol{\gamma}_m\}_{m=1}^{M}} \frac{1}{N}\sum_{t=1}^{N} L\Big(y_t, \sum_{m=0}^{M}\delta_m b(\mathbf{x}_{t-k};\boldsymbol{\gamma}_m)\Big). \tag{6}$$

This minimization problem can be approximated using forward stagewise additive modeling technique. This is done by adding new base learner functions to the expansion without altering the functions already included in the ensemble. At each step m the base learner function $b(\mathbf{x}_{t-k};\boldsymbol{\gamma}_m)$ which best fits the negative gradient of the loss function is selected and added to the ensemble. Using least squares as the fitting criterion while searching for the optimal base learner function leads to the general gradient boosting algorithm by Friedman (2001):

Algorithm 1 *Gradient boosting*

$F_0(\mathbf{x}_{t-k}) = \arg\min_{\rho} \frac{1}{N}\sum_{t=1}^{N} L\left(y_t, \rho\right)$

for $m \leftarrow 1$ *to* M ***do:***

$\tilde{y}_t = -\left.\frac{\partial L(y_t, F(\mathbf{x}_{t-k}))}{\partial F(\mathbf{x}_{t-k})}\right|_{F(\mathbf{x}_{t-k})=F_{m-1}(\mathbf{x}_{t-k})}, t = 1,\ldots,N$

$\boldsymbol{\gamma}_m = \arg\min_{\boldsymbol{\gamma},\delta} \sum_{t=1}^{N}\left[\tilde{y}_t - \delta b(\mathbf{x}_{t-k};\boldsymbol{\gamma})\right]^2$

$\rho_m = \arg\min_{\rho} \sum_{t=1}^{N} L\left(y_t, F_{m-1}(\mathbf{x}_{t-k}) + \rho b(\mathbf{x}_{t-k};\boldsymbol{\gamma}_m)\right)$

$F_m(\mathbf{x}_{t-k}) = F_{m-1}(\mathbf{x}_{t-k}) + \rho_m b(\mathbf{x}_{t-k};\boldsymbol{\gamma}_m)$

end for

Friedman (2001) suggests a slight modification to Algorithm 1 when regression trees are used as the base learner function. Regression trees are a simple yet powerful tool that partition the feature space into a set of J non-overlapping rectangles and attach a simple constant to each one. The base learner function of a J-terminal node regression tree can be written as

$$b(\mathbf{x}_{t-k};\{c_j, R_j\}_{j=1}^{J}) = \sum_{j=1}^{J} c_j I(\mathbf{x}_{t-k} \in R_j), \tag{7}$$

where the functional estimate is a constant c_j in region R_j. According to Friedman (2001), the additive J-terminal node regression tree in equation (7) can be seen as a combination

of J separate base learner functions. One base learner for each terminal node of the regression tree. Therefore after estimating the terminal node regions $\{R_{jm}\}_{j=1}^{J}$ at the mth iteration with least squares on line 4 of the Algorithm 1 the line search step on line 5 should produce separate estimates for each terminal node of the regression tree. This minimization problem can be written as

$$\{\hat{c}_{jm}\}_{j=1}^{J} = \underset{\{c_j\}_{j=1}^{J}}{\arg\min} \sum_{t=1}^{N} L\Big(y_t, F_{m-1}(\mathbf{x}_{t-k}) + \sum_{j=1}^{J} c_j I(\mathbf{x}_{t-k} \in R_{jm})\Big). \tag{8}$$

The ensemble update on the last line of Algorithm 1 is then a sum of these J terminal node estimates obtained in equation (8)

$$F_m(\mathbf{x}_{t-k}) = F_{m-1}(\mathbf{x}_{t-k}) + \sum_{j=1}^{J} \hat{c}_{jm} I(\mathbf{x}_{t-k} \in R_{jm}).$$

2.2 Cost-sensitive gradient boosting with class weights

With a high class imbalance there is a risk that the estimated binary classifier is skewed towards predicting the majority class well (He and Garcia, 2009). An algorithm can be made cost-sensitive by weighting the dataspace according to the misclassification costs (Branco, Torgo and Ribeiro, 2016). This weighting approach is sometimes referred to as rescaling in the previous literature (see e.g., Zhou and Liu, 2010). The asymmetric misclassification costs, which are the building block of cost-sensitive learning, are incorporated to the gradient boosting model by introducing a binary class weight for each observation in the data. In the traditional gradient boosting model the sample counterpart of the loss function is the sample mean and the minimization problem can be written as in equation (6). By introducing a vector of class weights we end up minimizing the weighted average of the sample loss function

$$\min_{\{\delta_m, \boldsymbol{\gamma}_m\}_{m=1}^{M}} \frac{1}{\sum_{t=1}^{N} w_t} \sum_{t=1}^{N} w_t L\Big(y_t, \sum_{m=1}^{M} \delta_m b(\mathbf{x}_{t-k}; \boldsymbol{\gamma}_m)\Big). \tag{9}$$

If the weights w_t are equal for each observation the weighted average in equation (9) reduces to the sample mean.

Elkan (2001) suggests weighting the minority class observations according to the ratio in misclassification costs. Suppose c_{10} denote the cost when we fail to predict a recession and c_{01} when we give a false alarm of recession. The optimal weight for the minority class observations is then

$$w^* = \frac{c_{10}}{c_{01}}. \tag{10}$$

In many cases the exact misclassification costs are unknown and we must rely on rules such that misclassifying the minority class is more costly (Maloof, 2003). The class weights are basically arbitrary as they depend on the unknown preferences how harmful different types of misclassification is considered to be. In this paper we use the data-based approach by Zhou (2012) and choose the weights according to the class imbalance observed in the dataset

$$w_t = \begin{cases} \frac{\sum_{t=1}^{N}(1-y_t)}{\sum_{t=1}^{N} y_t}, & \text{if } y_t = 1 \\ 1, & \text{if } y_t = 0 \end{cases} . \tag{11}$$

As can be seen from equation (11) the weights depend on the ratio of the number of datapoints in both classes. These binary weights ensure that the sum of weights are equal in both classes. The aim of choosing these weights is to force the algorithm to provide a balanced degree of predictive accuracy between the two classes.

The cost-sensitive gradient boosting algorithm with class weights follows the steps described in Algorithm 1 but the binary class weights can have an effect on each step of the algorithm. Table 1 illustrates how the class weights alter different parts of the gradient boosting algorithm, when J-terminal node regression trees are used as the base learner functions and the loss function to be minimized is the binomial deviance.

Table 1: The effect of class weights on the gradient boosting algorithm

Step	**Value**
Loss function	$\frac{-2\sum_{t=1}^{N} w_t[y_t F(\mathbf{x}_{t-k}) - \log(1+e^{F(\mathbf{x}_{t-k})})]}{\sum_{t=1}^{N} w_t}$
Initial value	$F_0(\mathbf{x}_{t-k}) = \log(\frac{\sum_{t=1}^{N} w_t y_t}{\sum_{t=1}^{N} w_t(1-y_t)})$
Gradient	$\tilde{y}_{tm} = y_t - p_t$, $where$ $p_t = \frac{1}{1+e^{-F_{m-1}(\mathbf{x}_{t-k})}}$
Split criterion	$i^2(R_l, R_r) = \frac{W_l W_r}{W_l + W_r}(\bar{g}_l - \bar{g}_r)^2$, $W_l = \sum_{x_{t-k} \in R_l} w_t$, $\bar{g}_l = \frac{1}{W_l}\sum_{x_{t-k} \in R_l} w_t \tilde{y}_{tm}$
Terminal node estimate	$\hat{c}_{jm} = \frac{\sum_{x_{t-k} \in R_j} w_t(y_t - p_t)}{\sum_{x_{t-k} \in R_j} w_t p_t(1-p_t)}$

Note that the values for each step of the ordinary gradient boosting model can be obtained from Table 1 by setting all the weights equal to one. The cost-sensitive and the traditional gradient boosting algorithms differ starting from the initial values. As the first gradient vector is based on the initial value the gradients are also different. The biggest differences between these two algorithms however are related to the estimation of the regression tree base learners at each iteration m of the algorithm. Blagus and

Lusa (2017) argue that the class imbalance problem of the gradient boosting model with high-dimensional data is related to the inappropriately defined terminal regions R_j.

Next we will consider how class weights can have an effect on both the estimated terminal node regions and the terminal node estimates of the regression tree base learner. When J-terminal node regression tree is used as the base learner function, the $J-1$ recursive binary splits into regions R_l and R_r dividing the predictor space into J non-overlapping terminal node regions $\{R_j\}_{j=1}^{J}$ are obtained by maximizing the least-squares improvement criterion. These splits are based on a slightly different criterion if class weights are used. For this reason the estimated terminal node regions and the terminal node estimates can be different between the two algorithms.

From Table 1 we can see how the split criterion is based on two parts. The first part $\frac{W_l W_r}{W_l+W_r}$ illustrates how each split into regions R_l and R_r in cost-sensitive gradient boosting is based on the sum of weights in these two categories instead of the number of observations. The latter part of the split criterion $(\bar{g}_l - \bar{g}_r)^2$ shows that instead of the average gradient we compare the weighted average of the gradient in the regions, when searching for the optimal split point. From the last row in Table 1 one can note how the terminal node estimates are functions of both the terminal node regions and the class weights itself and hence the final estimates can be different between the two algorithms.

2.3 Regularization parameters in gradient boosting

Friedman (2001, 2002) introduces several add-on requlariza tion techniques to reduce the risk of overfitting or to improve the overall performance of the gradient boosting algorithm. The parameters related to these techniques are often called tuning parameters since it is up to the user to finetune the parameter values for the particular problem at hand. Tuning parameters with the gradient boosting technique can be divided into two categories: parameters related to the overall algorithm and parameters related to the chosen base learner function.

Friedman (2001) incorporates a simple shrinkage strategy to slow down the learning process. In this strategy each update of the algorithm is scaled down by a constant called learning rate. The ensemble update on the last line of Algorithm 1 can then be written as

$$F_m(\mathbf{x}_{t-k}) = F_{m-1}(\mathbf{x}_{t-k}) + \upsilon \rho_m b(\mathbf{x}_{t-k}; \boldsymbol{\gamma}_m),$$

where $0 < \upsilon \leq 1$ is the learning rate. Learning rate is a crucial part of the gradient boosting algorithm as it controls the speed of the learning process by shrinking each gradient descent step towards zero. Friedman (2001) suggests to set the learning rate small enough for better generalization ability. Bühlmann and Yu (2010) reach a similar conclusion.

Breiman (1996) notes that introducing randomness when building each tree in an

ensemble can lead to substantial gains in prediction accuracy. Based on these findings Friedman (2002) develops stochastic gradient boosting in which subsampling is used to enhance the generalization ability of the gradient boosting model. At each round of the algorithm a random subsample of datapoints is drawn without replacement and the new base learner function is fitted using this random subsample. Simulation studies show that subsampling fraction around one half seems to work best in most cases (Friedman, 2002).

The total amount of iterations M needed however moves in the opposite direction to learning rate and subsampling. Gradient boosting is a flexible technique which can approximate basically any kind of functional form with sufficient amount of data. This flexibility can also come with a cost. Overfitting the training data is a risk that must be taken into consideration as it can lead to decreased generalization ability of the model. The optimal amount of iterations is usually chosen with early stopping methods such as using an independent test set or cross-validation.

When the amount of observations is scarce K-fold cross-validation is often the only alternative since we can not afford to set aside an independent test set. K-fold cross-validation is based on splitting the data into K non-overlapping folds. Each of these folds is used as a test set once while the model is estimated using the remaining $K-1$ folds. To reduce the effect of randomness the K-fold cross-validation process can be repeated R times (Kim, 2009). In the repeated K-fold cross-validation approach the estimate for the optimal stopping point is based on the average validation error produced by the K folds at each of these R repeats.

Instead of the traditional repeated K-fold we use a more conservative cross-validation approach since the risk of overfitting the data in the high-dimensional setup is fairly high. In this conservative approach only the validation error produced by the fold, which first reaches its minimum and therefore first starts to show signs of overfitting, is selected out of the K folds at each repetition. By denoting the found "weakest" fold in repetition r as k_r^*, the number of observations in this fold as $N_{k_r^*}$ and the model estimated without this fold as $\hat{F}^{-k_r^*}(\mathbf{x}_{t-k})$ the conservative cross-validation estimate for the prediction error can be written as

$$CV = \frac{1}{R}\sum_{r=1}^{R}\frac{1}{N_{k_r^*}}\sum_{t=1}^{N_{k_r^*}} L\big(y_t, \hat{F}^{-k_r^*}(\mathbf{x}_{t-k})\big), \tag{12}$$

where binomial deviance is used as the loss function $L(\cdot)$. The final estimate for the amount of iterations is the point where the estimated prediction error in (12) reaches its minimum. To the best of our knowledge this simple conservative approach has not been used in the previous academic research.

The complexity of the regression tree base learners is controlled by the number of

terminal nodes J in each regression tree. The amount of inner nodes $(J-1)$ in the regression tree limit the potential amount of interaction between predictors as shown with the ANOVA expansion of a function

$$F(\mathbf{x}_{t-k}) = \sum_j f_j(x_j) + \sum_{j,k} f_{jk}(x_j, x_k) + \sum_{j,k,l} f_{jkl}(x_j, x_k, x_l) + \cdots . \tag{13}$$

The simplest regression tree with just two terminal nodes can only capture the first term in equation (13). Higher order interactions are needed to be able to capture the latter terms, which are functions of more than one variable. These higher-order interactions require deeper trees. Hastie, Tibshirani and Friedman (2009) argue that trees with more than ten terminal nodes are seldom needed with boosting.

3 Results

3.1 Data and model setup

The dataset used in the empirical analysis is the FRED-MD monthly dataset. The selected timespan covers the period from January 1962 to June 2017. After dropping out variables that are not available for the full period the FRED-MD dataset consists of 130 different economic and financial variables related to different parts of the economy.[1] Three different forecasting horizons h are studied in the empirical analysis: short ($h = 3$), medium ($h = 6$) and long ($h = 12$).

All the available lag lengths k of the predictors up to 24 months are considered as potential predictors (assuming $k \geq h$). The total amount of predictors in the dataset take the value of 2860, 2470 or 1690 depending on the length of the forecasting horizon. For example, the total amount of predictors with the shortest forecasting horizon is 2860, which includes 22 different lags of these 130 variables. See Christiansen, Eriksen and Møller (2014) for a similar study where each lag is considered as a separate predictor.

The term spread has been noted as the best single predictor of recessions and economic growth in general in the U.S. (see e.g., Dueker, 1997; Estrella and Mishkin, 1998; Wohar and Wheelock, 2009). To see if it is actually worthwhile to go through these huge predictor sets with the gradient boosting models, we use a simple logit model with the term spread as a benchmark model. Kauppi and Saikkonen (2008) note that setting the lag length k equal to the forecasting horizon h may not be optimal in all cases. To take this into account we introduce the six nearest lag lengths of the term spread as additional predictors. The term spread is measured as the interest rate spread between

[1] All ISM-series (The Institute for Supply management) have been removed from the FRED-MD dataset starting from 2016/6. These series have been re-obtained using Macrobond. For more general information about the dataset see https://research.stlouisfed.org/econ/mccracken/fred-databases/

the 10-year government bonds and the effective federal funds rate as this is included in the FRED-MD dataset.

The estimated conditional probabilities for different models are evaluated using the receiver operating characteristic curve (ROC). The area under the ROC-curve (AUC) measures the overall classification ability of the model without restricting to a certain probability threshold. AUC-values closer to one indicate better classification ability whereas values close to one half are no better than a simple coin toss. For a more comprehensive review of the AUC-measure in economics context see e.g., Berge and Jorda (2011) and Pönkä and Nyberg (2016).

The gradient boosting model involves internal model selection as the regression trees selected at each step of the algorithm may be functions of different predictors. Some predictors are chosen more often than others and can be considered more important. Breiman et al. (1984) introduce a measure for the relevance of a predictor x_p in a single J-terminal node regression tree T

$$\hat{I}_p^2(T) = \sum_{j=1}^{J-1} \hat{\imath}_j^2 I(v_j = p), \tag{14}$$

where v_j is the splitting variable of inner node j and $\hat{\imath}_j^2$ is the empirical improvement in squared error as a result of this split. The least squares improvement criterion was introduced in Table 1.

The measure in equation (14) is based on a single tree, but it can be generalized to additive tree expansions as well (Friedman, 2001). The relative influence of a variable x_p for the entire gradient boosting ensemble is simply an average over all the trees $\{T_m\}_{m=1}^M$ in the ensemble

$$\hat{I}_p^2 = \frac{1}{M} \sum_{m=1}^{M} \hat{I}_p^2(T_m). \tag{15}$$

The relative influence measure in equation (15) is used to illustrate the most important recession indicators with the gradient boosting model. The relevance of a predictor x_p in the recursive out-of-sample forecasting is the average $\hat{I}_p^2$ of the estimated models.

The following results are obtained using the R programming environment for statistical computing (R Core Team, 2017). The GBM-package (Ridgeway, 2017) with bernoulli loss function is used to estimate the gradient boosting models. With such huge predictor sets it is likely that there are interaction between some predictors. For this reason the maximum tree depth is set to 8 leading to regression trees with nine terminal nodes. Döpke et al. (2017) use 6-terminal node regression trees while predicting recessions in Germany with a much smaller predictor set.

The minimum number of observations required in each terminal node of a regression tree is set to one allowing the tree building process to be as flexible as possible. Similar

results are obtained when setting the minimum number of observations to five as is used by Döpke et al. (2017).[2] Learning rate is set to a low value of 0.005 and the default value of 0.5 is used as the subsampling fraction. The conservative cross-validation approach presented in equation (12) is conducted using 5 folds and 5 repeats throughout this research to find the optimal amount of iterations. In order to keep the computational time feasible the maximum amount of iterations is set to 800.

3.2 In-sample results

Three different models are compared in the in-sample analysis using the full dataset. The benchmark model (bm) is a simple logit model with seven lags of the term spread as predictors. GBM is the ordinary gradient boosting model and wGBM stands for the cost-sensitive gradient boosting model with class weights. The class weights are formed according to equation (11). The binary response variable for each model is the business cycle chronology provided by the NBER.

Table 2 summarizes the in-sample performance as measured with the area under the ROC-curve (AUC) of these three models for all the different forecasting horizons. The rows of the table present the different models and the columns stand for the forecasting horizons considered. The validation AUCs from the 5-fold cross-validation repeated five times are reported in parenthesis.

Table 2: In-sample AUC (1962/01 - 2017/06)

	Forecast horizon, Months		
Model specification	3	6	12
Benchmark	0.890 (0.881)	0.910 (0.902)	0.914 (0.897)
GBM	1.000 (0.985)	1.000 (0.980)	1.000 (0.956)
wGBM	1.000 (0.987)	1.000 (0.981)	1.000 (0.961)

As expected, the non-linear gradient boosting models do a better job forecasting recessions in-sample. The larger information set and the more flexible functional form of the GBM-models allow for a more detailed in-sample fit. The perfect in-sample AUCs for the GBM-models can raise questions of overfitting. As a result of using these moderate sized regression trees as base learner functions the GBM-models achieve nearly perfect classification ability after only a few iterations. This can be confirmed by training a shallow single decision tree to the full dataset. The single decision tree alone is sufficient to produce very high in-sample AUCs, even after restricting the predictor

[2] Results upon request.

space to consider only the eight different interest rate spreads (and their lag lengths).[3] Thereby it is not completely surprising that an ensemble of trees yield a perfect in-sample fit as measured with AUC. For example, the cost sensitive GBM-model with the shortest forecasting horizon reaches an AUC value of 0.997 after just five iterations. However it should be noted that the estimated conditional probabilites at this point range between 0.488 and 0.512, values that are only slightly different from the initial value of one half because of the shrinkage strategy described in Section 2.3. It could be argued that the AUC may not be the most suitable criterion when evaluating the in-sample performance in this setup. But since the main emphasis is on the out-of-sample performance of the models the AUCs are reported here for comparison.

The validation AUCs reported in Table 2 provide additional insight into the potential overfitting problem since large deviations between the in-sample and validation performance is typically seen as a sign of overfitting. The validation AUCs for the GBM-models are of similar magnitude as the in-sample AUCs and therefore do not indicate overfitting. Döpke et al. (2017) also report validation AUCs close to one when forecasting recessions in Germany with the gradient boosting model. The validity of the traditional random sampling techniques used in cross-validation with such a highly autocorrelated binary response variable should be further examined. This however is beyond the scope of this research.

Table 2 shows how the cost sensitive GBM-model outperforms the other two models as measured with the validation AUC, although the difference between the two GBM-models is small. The gap in validation AUCs between the benchmark and GBM-models decreases slightly as the forecasting horizon grows. Graphical illustrations are an important part of recession forecasting since these can give a better picture of the false alarms and other potential problems related to the models. The estimated conditional probabilities that the economy is in recession h-months from now are calculated according to equation (2). These in-sample estimated conditional probabilites are illustrated in Figure 1 for each of the three models and forecasting horizons.

[3] The in-sample AUCs with a single decision tree are close to or well above 0.95 depending on the forecasting horizon. On the other hand, restricting the GBM-models by considering only the simplest stump regression trees and / or only the interest rate spreads as predictors are not sufficient as models produce in-sample AUCs of one or really close to it. Results upon request.

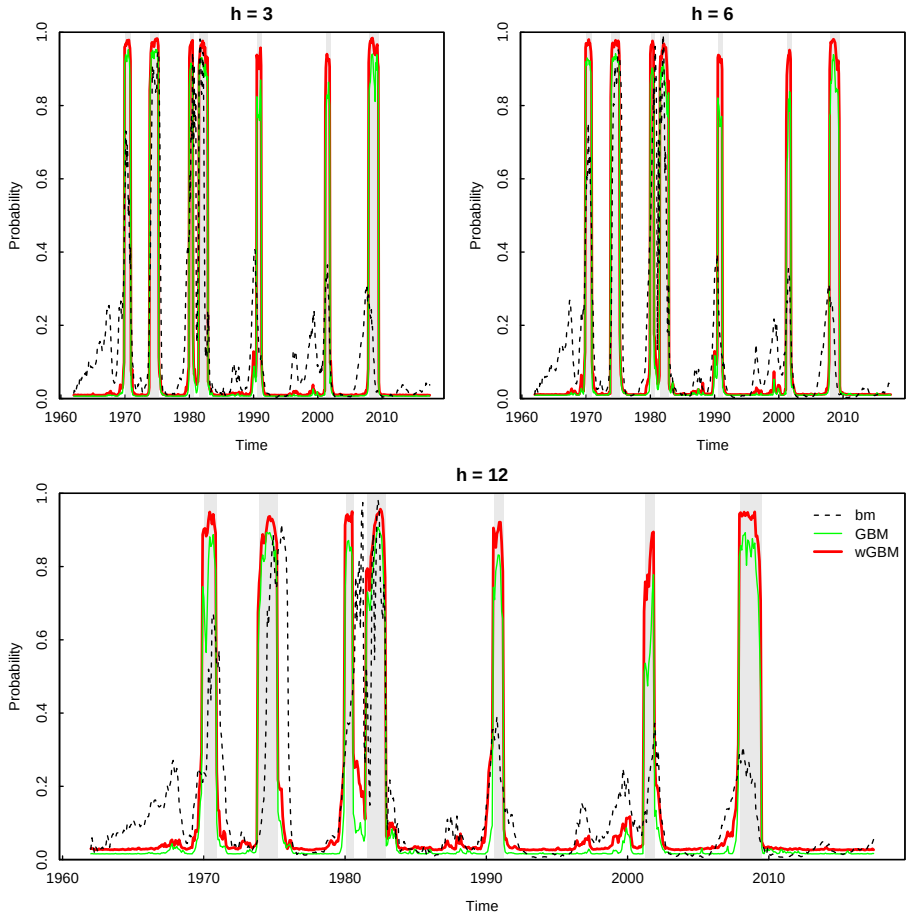

Figure 1: In-sample estimated conditional probabilities

The conditional probabilities for both GBM-models can be seen to mimic the shaded recession periods quite nicely. The in-sample fits for the two GBM-models have a rather similar shape without any major differences, which is in line with the results in Table 2. However, the recession signals produced by the cost-sensitive GBM-model are constantly stronger compared to the other two models with all the forecasting horizons. It is also noteworthy how the benchmark logit model produces a lot weaker signals for the last three recessions compared to the GBM-models. Figure 1 also shows how the estimated conditional probabilities for the GBM-models are not exactly zero or one and the in-sample fit is not perfect in probability terms. Using forecast performance evaluation criterion other than AUC, such as the binomial deviance or the quadratic probability score, would not indicate perfect in-sample fit.

3.3 Out-of-sample results

Good in-sample results may not always reflect the out-of-sample predictive ability of the model. An expanding window forecasting procedure is used to examine the true predictive ability of the models. Both Berge (2015) and Ng (2014) use rolling window when forecasting U.S. recessions. To ensure the maximum sample size for the estimation

of each model an expanding window approach is used in this study.

The out-of-sample evaluation period covers the period starting from December 1988 to June 2017. Because of high computational cost the GBM-models are re-estimated only once a year in December. The class weights are updated according to equation (11) as the proportion of zeros and ones change for the binary response. The business cycle recession and expansion periods are not available in real time. The publication lag of the NBER business cycle chronology is thus assumed to be 12 months.

The results from the recursive out-of-sample forecasting procedure are reported in Table 3. The out-of-sample performance as measured with the area under the ROC-curve is illustrated for the different models at each of the three forecasting horizons.

Table 3: Out-of-sample AUC (1988/12 - 2017/06)

	Forecast horizon, Months		
Model specification	3	6	12
Benchmark	0.748	0.811	0.919
GBM	0.841	0.816	0.867
wGBM	0.915	0.861	0.928

The out-of-sample AUCs show that the cost-sensitive GBM-model outperforms the other two models with all the forecasting horizons. The difference in AUCs between the traditional and cost-sensitive gradient boosting models are quite similar with all the forecasting horizons. The average difference of the AUCs between the two GBM-models is 0.06.

The out-of-sample performance for the traditional GBM-model is quite heavily deteriorated when compared to the in-sample AUCs reported in Table 2. The standard GBM-model can outperform the benchmark model only at the shortest forecasting horizon. This diminished out-of-sample forecasting ability of the traditional GBM-model could indicate problems related to the class imbalance of the response. Blagus and Lusa (2017) note that the traditional GBM-model can perform poorly on high-dimensional data with class imbalance. Figure 2 illustrates the out-of-sample estimated conditional probabilities calculated according to equation (2) for all the different forecasting horizons and models.

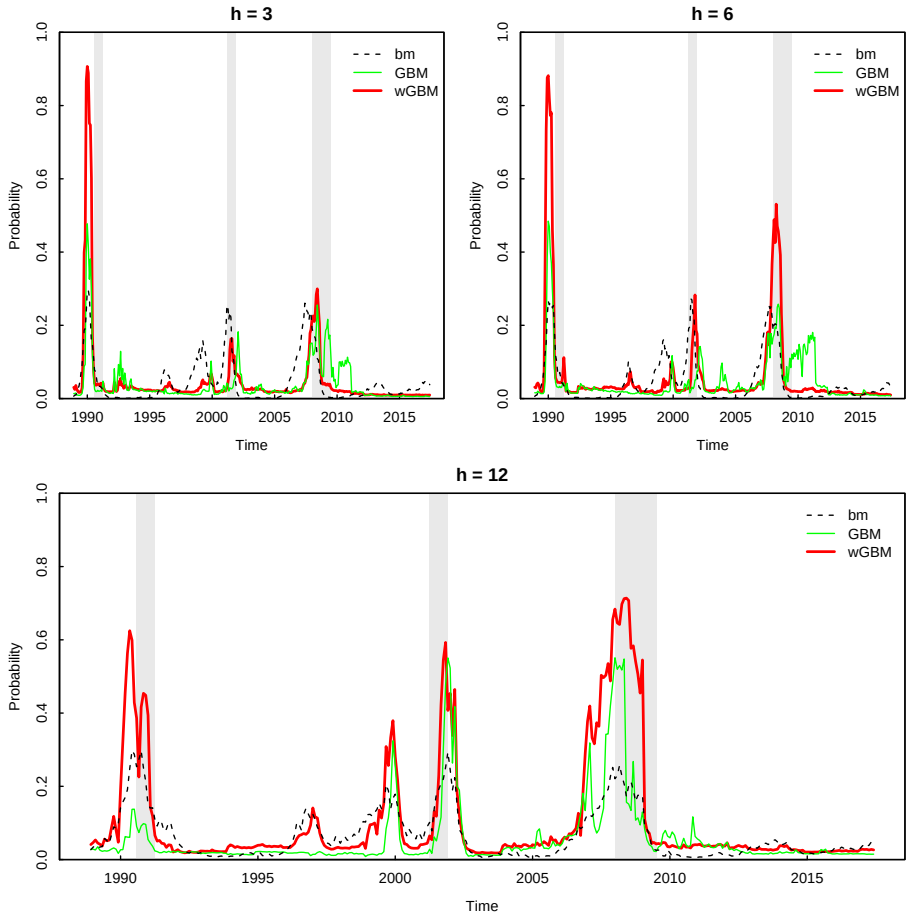

Figure 2: Out-of-sample estimated conditional probabilities

Figure 2 shows how the recession probabilities for each of the models with the short and medium term forecasting horizons spike just before the actual recession period in the early ninetees. Although these spikes are considered as false alarms and decrease the out-of-sample performance of the models, this hightened risk of an upcoming recession could have considerable practical importance.

Figure 2 also illustrates the problems related to the diminished out-of-sample performance of the traditional GBM-model. The traditional GBM-model provides several false alarms, especially at the short and medium term forecasting horizons. With the longest forecasting horizon the traditional GBM-model give a rather weak signal of the upcoming recession period in the early ninetees when compared to the other two models.

The cost-sensitive GBM-model on the other hand provides clear warnings of the upcoming recession periods in the short and medium term without any major false alarms. Although the recession signal for the second recession period with the shortest forecasting horizon is quite modest. It should be noted that the magnitude of the recession signals are diminished for each of the three models when compared to the in-sample probabilities in Figure 1.

With the 12-month forecasting horizon the cost-sensitive GBM-model provides strong warning signals for each of the three recessions. The estimated recession probabilities of the cost-sensitive GBM-model bears a close resemblance to the benchmark model. This also includes the two false alarms that are typical when predicting recessions with the term spread (see e.g., Kauppi and Saikkonen, 2008; Nyberg, 2010).

To further consider the composition of the estimated cost-sensitive GBM-models Table 4 presents the ten most important out-of-sample predictors according to the relative influence measure presented in equation (15).

Table 4: Top-10 out-of-sample predictors for wGBM

$h = 3$		$h = 6$		$h = 12$	
Variable	Rel.inf	Variable	Rel.inf	Variable	Rel.inf
6mth - FFrate_4	5.464	6mth - FFrate_6	8.722	10yr - FFrate_12	18.051
10yr - FFrate_9	4.920	10yr - FFrate_9	4.979	5yr - FFrate_15	6.347
6mth - FFrate_6	4.744	5yr - FFrate_15	4.026	5yr - FFrate_14	3.634
6mth - FFrate_5	4.581	1yr - FFrate_6	3.941	10yr - FFrate_13	2.778
6mth - FFrate_7	3.103	6mth - FFrate_7	3.739	5yr - FFrate_16	2.466
5yr - FFrate_15	2.988	3mth - FFrate_6	3.570	10yr - FFrate_14	2.400
10yr - FFrate_8	2.757	10yr - FFrate_8	3.111	5yr - FFrate_13	1.808
3mth - FFrate_6	2.337	1yr - FFrate_7	2.512	AAA - FFrate_12	1.787
1yr - FFrate_6	2.310	6mth - FFrate_8	2.175	5yr - FFrate_12	1.688
1yr - FFrate_7	2.254	10yr - FFrate_11	2.048	PERMITS_15	1.496

The cost-sensitive GBM-models rely heavily on different kinds of interest rate spreads as can be seen in Table 4. The only non-interest rate based predictor is the fifteenth lag of the new private housing permits variable (PERMITS_15) with the longest forecasting horizon. This is a bit surprising at the short and medium term forecasting horizons since variables describing the real economy are often found useful when predicting recessions with these forecasting horizons (see e.g., Berge, 2015). The heavy usage of interest rate spreads confirms that predictors with forecasting ability beyond the term spread are quite hard to find (see e.g., Estrella and Mishkin, 1998; Liu and Moench, 2016).

Models based on different kinds of interest rate spreads can be affected by the problems related to the predictive power of the term spread noted in the previous literature. Several studies show how the term spread forecast U.S. output growth less accurately after the mid 1980s (see e.g., Estrella, Rodrigues and Schich, 2003; Stock and Watson, 2003). The slightly lower out-of-sample AUCs reported in Table 3 for each of the three models, including the benchmark model, are in line with this finding.

Table 4 shows how the interest rate spread between the 6-month treasury bill and the effective federal funds rate with the fourth lag (6mth - FFrate_4) is the most important predictor when predicting recessions three months ahead. The same predictor with the sixth lag is the most important predictor with the medium term forecasting horizon. The composition of the top-10 out-of-sample predictors are quite similar between the short and medium term horizons.

The chosen lag lengths of the predictors with the short and medium term horizons can deviate quite substantially from the length of the forecasting horizon. For example, the spread between the 5-year treasury bond and the effective federal funds rate with the fifteenth lag (5yr - FFrate_15) is an important predictor with both of these horizons. Similar observation can be made with the spread between the 10-year treasury bond and the effective federal funds rate with the ninth lag (10yr - FFrate_9). With the longest forecasting horizon the term spread with lag length equal to twelve (10yr - FFrate_12) has a very strong impact on the models as measured with the relative influence. Such dominance of a single predictor is not found with the short and medium term horizons.

4 Conclusions

This paper introduces a new cost-sensitive gradient boosting model which can take into account the class imbalance of the binary response variable. The cost-sensitive gradient boosting model is applied to predicting binary U.S. recession periods with a high-dimensional dataset of financial and macroeconomic variables. The internal model selection of the cost-sensitive gradient boosting algorithm provides important information about the most useful recession indicators and chosen lag lengths with different forecasting horizons.

The empirical results show how the cost-sensitive extension to the gradient boosting model produces stronger and more stable recession forecasts for the U.S. with each forecasting horizon compared to the traditional gradient boosting model. A logit model based on the term spread is used as a benchmark model to see if the more complex gradient boosting models provide predictive power beyond the best known simple model. The cost-sensitive model outperforms the benchmark model with each forecasting horizon whereas the traditional gradient boosting model is able to outperform the benchmark only at the shortest forecasting horizon. Different kinds of interest rate spreads are the most important predictors, even with the short and medium term forecasting horizons. The term spread is the dominant predictor when forecasting recessions one year ahead.

The current research can be extended in several ways. First of all, the binary values for the class weights were chosen so that both the minority and the majority class receive similar attention in the learning process. Different choices for the class weights

could be further examined. Especially in cases where the class imbalance is even more radical. The cost-sensitive approach could also be extended to multinomial classification problems, where different types of class imbalance problems can emerge. There could be for example more than one minority class with a multinomial response variable. Introducing model dynamics is another potential area for future research. This would allow iterative forecasts to be used instead of the forecast horizon-specific forecasts as in this study.

References

Berge, T. J. (2015). Predicting recessions with leading indicators: Model averaging and selection over the business cycle. *Journal of Forecasting*, 34(6):455–471.

Berge, T. J. and Jordà, Ò. (2011). Evaluating the classification of economic activity into recessions and expansions. *American Economic Journal: Macroeconomics*, 3(2):246–77.

Blagus, R. and Lusa, L. (2013). Smote for high-dimensional class-imbalanced data. *BMC Bioinformatics*, 14(1):106.

Blagus, R. and Lusa, L. (2017). Gradient boosting for high-dimensional prediction of rare events. *Computational Statistics & Data Analysis*, 113:19 – 37.

Branco, P., Torgo, L., and Ribeiro, R. (2016). A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys*, 49.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. J. (1984). *Classification and Regression Trees*. Wadsworth, New York.

Bühlmann, P. and Yu, B. (2010). Boosting. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(1):69–74.

Christiansen, C., Eriksen, J. N., and Møller, S. V. (2014). Forecasting us recessions: The role of sentiment. *Journal of Banking & Finance*, 49:459 – 468.

Döpke, J., Fritsche, U., and Pierdzioch, C. (2017). Predicting recessions with boosted regression trees. *International Journal of Forecasting*, 33(4):745–759.

Dueker, M. J. (1997). Strengthening the case for the yield curve as a predictor of u.s. recessions. *Federal Reserve Bank of St. Louis Economic Review*, 79:41–51.

Elkan, C. (2001). The foundations of cost-sensitive learning. In *In Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 973–978.

Estrella, A. and Mishkin, F. (1998). Predicting u.s. recessions: Financial variables as leading indicators. *The Review of Economics and Statistics*, 80(1):45–61.

Estrella, A., Rodrigues, A. R., and Schich, S. (2003). How stable is the predictive power of the yield curve? evidence from germany and the united states. *The Review of Economics and Statistics*, 85(3):629–644.

Fan, W., Stolfo, S. J., Zhang, J., and Chan, P. K. (1999). Adacost: Misclassification cost-sensitive boosting. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML '99, pages 97–105, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, ICML'96, pages 148–156, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 28:337–407.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367 – 378.

Galar, M., Fernández, A., Tartas, E. B., Bustince, H., and Herrera, F. (2012). A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42:463–484.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference and prediction*. Springer, 2 edition.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.

Kauppi, H. and Saikkonen, P. (2008). Predicting u.s. recessions with dynamic binary response models. *The Review of Economics and Statistics*, 90(4):777–791.

Kim, J.-H. (2009). Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational Statistics & Data Analysis*, 53(11):3735 – 3745.

Liu, W. and Moench, E. (2016). What predicts us recessions? *International Journal of Forecasting*, 32(4):1138 – 1150.

Maloof, M. A. (2003). Learning when data sets are imbalanced and when costs are unequal and unknown. In *ICML-2003 Workshop on Learning from Imbalanced Data Sets II*.

Ng, S. (2014). Viewpoint: Boosting recessions. *Canadian Journal of Economics*, 47(1):1–34.

Nyberg, H. (2010). Dynamic probit models and financial variables in recession forecasting. *Journal of Forecasting*, 29(1-2):215–230.

Nyberg, H. and Pönkä, H. (2016). International sign predictability of stock returns: The role of the United States. *Economic Modelling*, 58(C):323–338.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Ridgeway, G. (2017). *gbm: Generalized Boosted Regression Models*. R package version 2.1.3.

Stock, J. H. and Watson, M. W. (2003). Forecasting output and inflation: The role of asset prices. *Journal of Economic Literature*, 41(3):788–829.

Sun, Y., Kamel, M. S., Wong, A. K., and Wang, Y. (2007). Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition*, 40(12):3358 – 3378.

Ting, K. M. (2000). A comparative study of cost-sensitive boosting algorithms. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pages 983–990, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Wheelock, D. C. and Wohar, M. E. (2009). Can the term spread predict output growth and recessions? a survey of the literature. *Federal Reserve Bank of St. Louis Review*, Part 1(Sep/Oct):419–440.

Zhou, Z.-H. (2012). *Ensemble Methods: Foundations and Algorithms*. Chapman & Hall/CRC, 1st edition.

Zhou, Z.-H. and Liu, X.-Y. (2010). On multi-class cost-sensitive learning. *Computational Intelligence*, 26(3):232–257.