Zieba, Maciej; Härdle, Wolfgang Karl

Working Paper

Beta-boosted ensemble for big credit scoring data

SFB 649 Discussion Paper, No. 2016-052

Provided in Cooperation with:
Collaborative Research Center 649: Economic Risk, Humboldt University Berlin

Suggested Citation: Zieba, Maciej; Härdle, Wolfgang Karl (2016) : Beta-boosted ensemble for big credit scoring data, SFB 649 Discussion Paper, No. 2016-052, Humboldt University of Berlin, Collaborative Research Center 649 - Economic Risk, Berlin

This Version is available at:
https://hdl.handle.net/10419/148888

SFB 649 Discussion Paper 2016-052

Beta-boosted ensemble for big credit scoring data

Maciej Zieba *
Wolfgang K. Härdle*²

* Wroclaw University of Science and Technology, Republic of Poland
*² Humboldt-Universität zu Berlin, Germany

This research was supported by the Deutsche Forschungsgemeinschaft through the SFB 649 "Economic Risk".

http://sfb649.wiwi.hu-berlin.de
ISSN 1860-5664

SFB 649, Humboldt-Universität zu Berlin
Spandauer Straße 1, D-10178 Berlin

Beta-boosted ensemble for big credit scoring data

Maciej Zięba, Wolfgang Karl Härdle

Abstract In this work we present a novel ensemble model for a credit scoring problem. The main idea of the approach is to incorporate separate beta binomial distributions for each of the classes to generate balanced datasets that are further used to construct base learners that constitute the final ensemble model. The sampling procedure is performed on two separate ranking lists, each for one class, where the ranking is based on prepotency of observing positive class. Two strategies are considered: one assumes mining easy examples and the second one forces good classification of hard cases. The proposed solutions are tested on two big datasets on credit scoring.

JEL classification: C53, Forecasting and Prediction Methods; Simulation

Key words: credit scoring, ensemble model, beta distribution, Beta boost, big data

Maciej Zięba
Wroclaw University of Science and Technology, Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland, e-mail: maciej.zieba@pwr.edu.pl

Wolfgang Karl Härdle
Professor at Humboldt-Universität zu Berlin, Ladislaus von Bortkiewicz chair of statistics and Director of C.A.S.E. - Center for Applied Statistics and Economics, Humboldt-Universität zu Berlin, Spandauer Straße 1, 10178 Berlin, Germany and School of Business, Singapore Management University, 50 Stamford Road, Singapore 178899. e-mail: haerdle@wiwi.hu-berlin.de

Financial support from the Deutsche Forschungsgemeinschaft via CRC *Economic Risk* and IRTG 1792 *High Dimensional Non Stationary Time Series*, Humboldt-Universität zu Berlin, is gratefully acknowledged.

1 Introduction

The problem of constructing a decision model to distinguish good and bad consumers can be defined as a dichotomous classification task, where the positive class (usually less numerous) represents "bad" applicants and the negative class stays behind "good" cases. Usually, instead of obtaining the binary classification result we aim at estimating the probability of credit repayment for each of the consumers. Basing on the probabilities the financial institution is capable to define the various profiles of the consumers. The common procedure for that kind of applications is to separate from training some group of labeled consumers and sort them according to the predictive probability using the trained model. The sorted group with the given labels is further used to distinguish the profiles. As a consequence, a higher patience is given to construct models that are characterized by good sorting capabilities than to the typical classifiers used for binary classification. Instead of maximizing the *accuracy of prediction* the community working on the credit scoring models aims at achieve the highest value of *AUC (area under ROC curve)* criterion that stays behind sorting capabilities of the models.

Various machine learning algorithms were applied to solve credit scoring and fraud detection problems, such as: neural networks [13, 18, 22], Gaussian Processes [10], various extensions of SVMs (Support Vector Machines) [2, 3, 4, 7, 8, 9, 23] or comprehensible models based on neural structures [20] or SVMs [16].

Ensemble methods have also gained particular attention in the field of credit scoring. The general idea of this type of models is based on constructing many component models (so called base learners) that are joined together as one complex classifier. Usually, the base model is so-called weak learner that is characterized by poor individual performance, but strong learners are also used for particular ensemble models. Authors of [17] present very beneficial comparison of the standard ensemble procedures in application to credit scoring tasks. Some more up-to-date analysis of this kind of models for this particular application were presented in [1] and [24]. The most recent models make use of various types of base learners [11], joined two strategies of diversification on features and data levels [15], switching class labels [25], boosting neural networks [21] or using ensemble of cost-sensitive SVMs trained with active learning strategy [26]. Most recent studies studies show the great benefit of using Extreme Boosted Trees [27].

Here, we aim at constructing a novel boosting approach that works independently on selected base model and performs well on big credit scoring datasets. The key idea of this approach is to apply a strategy to sample examples for each of the boosting iterations to construct the base learners. We make use of particular Beta Binomial distributions that are applied to the sorted training data according to the prediction probabilities returned by current ensemble model. In this work we distinguish two sampling strategies: the first strategy aims at sampling with the higher probability the examples that are already well located in the ranking. The other strategy is an example of so-called *hard examples* mining where the higher probabilities are given to the examples badly predicted and badly located in the ranking. Our approach was tested on the two benchmark datasets using two base models: Logis-

tic Regression and Decision Tree classifier. The results show that the first strategy works fine with the stable models like Logistic Regression, while the second strategy improves the quality of weak learners like Decision Trees.

The paper is organized as follows. In Section 2 we present the *BetaBoost* algorithm. In Section 3 we introduce some experimental studies investigating the performance of the approach. The paper is summarized with some conclusions presented in Section 4.

2 Method description

The main idea of the proposed approach is to create an ensemble model that makes use of re-sampling diversification technique in only to increase its sorting capabilities. To achieve the goal, each of the base learners is trained using re-sampled training data. The re-sampling procedure makes use of two particular beta binomial distributions (one for each class) that are used to generate indexes of examples that are going to be taken in the next boosting iteration. The crucial step in the training procedure is sorting the training data according to predictive capabilities of the so far created ensemble model. As a consequence, the examples with higher probability value have higher indexes and are going to be selected more often in training iterations. For the sampling sampling procedure we propose to use *Beta Binomial* distribution which is going to be characterized in the next subsection.

2.1 Beta Binomial distribution

The beta binomial distribution is selected because it is capable to assign high probabilities to particular regions of the sorted data according to predictive probability values of the training examples. Practically, it means that we are capable to concentrate our model either on learning from difficult-to-distinguish credit consumers, or put the higher impact on learning from the easy-to-classify client applicants.

The flexibility of beta binomial distribution is controlled by three parameters:

- Shape parameters a and b that are characteristic for beta distribution ($a, b > 0$).
- Parameter N that represents the number of trials characteristic for binomial distribution ($N \in \mathbf{N_0}$).

The probability function for beta binomial distribution ($BBin(a,b,N)$) can be presented in the following form:

$$p(k;a,b,N) = \binom{N}{k} \frac{B(k+a, N-k+b)}{B(a,b)}, \tag{1}$$

where $B(a,b)$ is the beta function. The plots of the probability function for various values of the parameters a and b are presented in Figure 1.

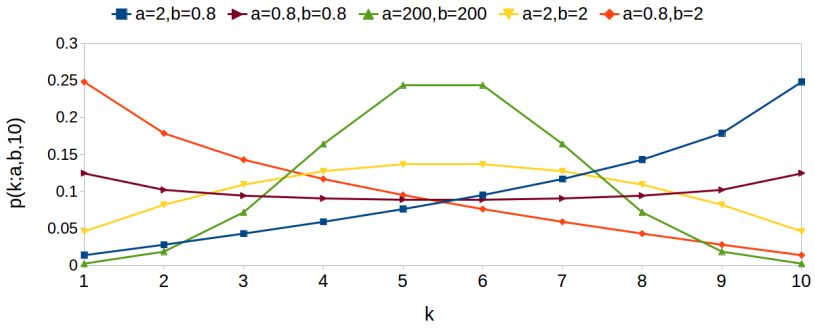


Fig. 1: Probability mas function for beta binomial distribution considering various a, b values.

The presented distribution has the important property for the particular values of shape parameters a and b. In this application we are concentrating on particular families of beta binomial distributions:

- The subset of distributions, where $a \leq 1$, $b \geq 1$ and $a \neq b$. If $k_1 > k_2$, then $p(k_1;a,b,N) < p(k_2;a,b,N)$.
- The subset of distributions, where $a \geq 1$, $b \leq 1$ and $a \neq b$. If $k_1 > k_2$, then $p(k_1;a,b,N) > p(k_2;a,b,N)$.

The selection of the particular distributions is indicated by the strategies that are going to be applied to train the ensemble model. For the first strategy we aim at putting the higher impact on selecting better located examples in the ranking so for the ranking list for negative examples (sorted according to probability of observing positive class) we apply the family of distributions that satisfies $p(k_1;a,b,N) < p(k_2;a,b,N)$, while for the ranking list for positive cases we use family of distribution that satisfies $p(k_1;a,b,N) > p(k_2;a,b,N)$. As a consequence it is more probable to select the examples properly located on the both of the lists.

For the second strategy we make use of the first family of distributions for positive ranking list and the second family for negative sorted samples. Contrary to previous strategy we aim at mining rather hard positive and negative examples and omitting well classified examples.

In the next section we present, how the beta binomial sampling is used in constructing the boosted model.

2.2 Beta-boosted ensemble model

In this work we aim at constructing the ensemble classifier for binary classification $y \in \{0,1\}$, composed of T base models:

$$p_T(y|\mathbf{x}) = \sum_{t=0}^{T} p_t p(y|\mathbf{x},t)^y \Big\{1 - p(y|\mathbf{x},t)\Big\}^{(1-y)}, \tag{2}$$

where $\mathbf{x}$ is vector of input features, $p(y|\mathbf{x},t)$ represents t-th base learner, and p_t is prior distribution over base learners.

For further work we assume that base learners are characterized by uniform distribution, so we can present the ensemble model given by equation (2) in the following form:

$$p_T(y=1|\mathbf{x}) = \frac{1}{T+1} \sum_{t=0}^{T} p(y=1|\mathbf{x},t). \tag{3}$$

We are interested in obtaining probability value for a given positive class therefore we will further operate on probability for this class, $p(y=1|\mathbf{x})$.

For the given predictor $p(y=1|\mathbf{x})$ and the set of examples $X_N = \{\mathbf{x}_n\}_{n=1}^N$ we can define the rank function $h(\mathbf{x}, X_N, p)$:

$$h(\mathbf{x}, X_N, p) = \sum_{n=1}^{N} \mathbb{I}\Big\{p(y=1|\mathbf{x}) > p(y=1|\mathbf{x_n})\Big\} \tag{4}$$

The procedure for creating the ensemble classifier can be described by Algorithm 1. To create the classifier we make use of training data $D_N = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$, that contains N training examples: N_1 positive and N_0 negative instances. We aim at constructing the ensemble model given by the equation (3).

Algorithm 1: BetaBoost

Input: Training data: $D_N = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$
Output: Ensemble model: $p_T(y=1|\mathbf{x})$ (see eq. (3))
Parameters: $BBin_0(\cdot)$ parameters for negative class: a_0, b_0,
$BBin_1(\cdot)$ parameters for positive class: a_1, b_1,
number of base learners: $T+1$.

Set $\mathbf{X}_{N_1} = \{\mathbf{x}_n : y_n = 1\}$ and $\mathbf{X}_{N_0} = \{\mathbf{x}_n : y_n = 0\}$;
Train weak learner $p(y|\mathbf{x},0)$ with data D_N;
for $t \leftarrow 1$ **to** T **do**
  Create ensemble predictor $p_{t-1}(y=1|\mathbf{x}) = \frac{1}{t+1}\sum_{j=0}^{t} p(y=1|\mathbf{x},j)$;
  Generate $\tilde{\mathbf{X}}_{N/2}^{(1)}$ with $sample(\mathbf{X}_{N_1}, p_{t-1}, a_1, b_1, N/2)$ (see Algorithm 2);
  Generate $\tilde{\mathbf{X}}_{N/2}^{(0)}$ with $sample(\mathbf{X}_{N_0}, p_{t-1}, a_0, b_0, N/2)$ (see Algorithm 2);
  Create new training data $\tilde{D}_N = (\tilde{\mathbf{X}}_{N/2}^{(1)}, 1) \cup (\tilde{\mathbf{X}}_{N/2}^{(1)}, 0)$;
  Train weak learner $p(y|\mathbf{x},k)$ with data $\tilde{D}_N$;
end

To initialize the training procedure we distinguish positive and negative examples denoting them by $\mathbf{X}_{N_1}$ and $\mathbf{X}_{N_2}$ respectively. We also initialize the ensem-

ble structure by training the first base learner $p(y|\mathbf{x},0)$ using initial training set $D_N = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$. In the next step we perform constructing the committee of T base classifiers in the training loop. Before creating the base learner we perform beta binomial sampling using separate distributions for each of the classes to obtain $N/2$ samples for each class. We use distributions for each of the classes, $BBin_0(a,b,N)$ to sample negatives and $BBin_1(a,b,N)$ to sample positives. We recommend to use particular families of distributions that was characterized in subsection 2.1.

Algorithm 2: Sampling procedure: $sample(X_N, p, a, b, N_{out})$

Input: Predictor $p(y=1|\mathbf{x})$, the set of examples $X_N = \{\mathbf{x}_n\}_{n=1}^N$, number of output samples N_{out}.
Output: Set of data samples $\tilde{X}_{Nout} = \{\mathbf{x}_n\}_{n=1}^{N_{out}}$
Parameters: $BBin(\cdot)$ parameters: a, b.

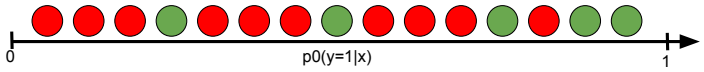
$\tilde{X}_0 \leftarrow \emptyset$;
for $n \leftarrow 1$ **to** N_{out} **do**
  Sample $k \sim BBin(a,b,N-1)$;
  $\tilde{X}_n \leftarrow \tilde{X}_{n-1} \cup \{\mathbf{x} \in X_N : h(\mathbf{x}, X_N, p) = k\}$, $h(\mathbf{x}, X_N, p)$ is given by eq. (4);
end

The procedure of sampling the data makes use of the currently created ensemble model $p_{t-1}(y=1|\mathbf{x})$ to determine the ranking position of the example $\mathbf{x}$ in the given set X_N using ranking function $h(\mathbf{x}, X_N, p)$ given by equation (4). The sampling procedure is performed independently for each of the classes and is described by Algorithm 2. First, we sample the integer k from $BBin(a,b,N-1)$ distribution. Second, we identify the sample that has ranking value equal to the sampled k value and include it into the set of output samples $\tilde{X}_n$. The sampling procedure is repeated N_{out} times to obtain the output set of examples, $\tilde{X}_{N_{out}}$. The procedure is equivalent to sorting the given data according to the given predictions and then sampling their position with beta binomial distribution.

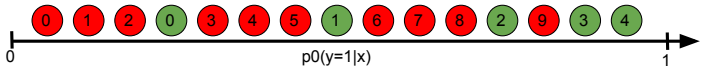
The sampling procedure is performed separately for the sets of positive and negative examples $\mathbf{X}_{N_1}$, $\mathbf{X}_{N_0}$ and, as a consequence, the new sets $\tilde{\mathbf{X}}_{N/2}^{(1)}$ and $\tilde{\mathbf{X}}_{N/2}^{(0)}$ are created and each of them contains $N/2$ sampled examples. The two set are then labeled and concatenated to the new training data $\tilde{D}_N$ that is further used to train the k-th base learner $p(y|\mathbf{x},k)$. The procedure is repeated T times to obtain ensemble model composed of $T+1$ base learners.

2.3 Toy example

Consider the toy example in which we have set of 15 examples, 5 from positive class and 10 from negative class. Assume, that we have the committee of the models that sorted the training examples according to the predictive probability $p(y=1|\mathbf{x})$ (see Figure 2a). Further, we assign individual ranking position for each of the considered



(a) Sorted data points according to the predictive distribution.



(b) Sorted data points with individual rankings for each class.

Fig. 2: The set of data examples sorted according to $p_0(y=1|\mathbf{x})$. Red circles represent negative examples, and green circles stand behind positive cases.

classes (see Figure 2b). Next, we assume individual Beta binomial distribution for each of the classes:

- $BBin_0(0.8,2,9)$ for negative examples.
- $BBin_1(2,0.8,4)$ for positive examples.

The selected distributions are consistent with the first strategy described in subsection 2.1, where we aim at mining easy examples from both classes. We take arbitrary values of the parameters ($a_0 = 0.8$, $b = 2$, $a = 2$, $b = 0.8$) just to illustrate the proposed algorithm. Considering real applications, the selection of the a and b is crucial for the training procedure. If the both values are close to 0 the distribution approaches uniform distribution, while for large a and small b examples with high positions are going to be selected multiple times. To select proper parameters for the distributions model selection procedure should be applied.

If we assume equal prior probabilities for selecting examples from minority and majority class, the sampling distribution for the next boosting iteration is presented in Figure 2.

If perform sampling with replacement from the given distribution we can obtain the set of examples that should be taken into next boosting iteration that is presented in Figure 4a. After learning the second base learner $p(y=1|x,1)$ and adding it to the ensemble model $p_1(y=1|\mathbf{x}) = \frac{p(y=1|x,0)+p(y=1|x,1)}{2}$ we obtain the better sorting of the data (see Figure 4b).

If we consider the *AUC* criterion (*area under ROC curve*) that represents the quality of the sorting capabilities for the binary classification models it increases from 0.76 to 0.92.

The idea that stays behind the proposed procedure is a proper selection of the sampling distributions the satisfy the conditions that are described in subsection 2.1. In this variant we take the distribution for sampling positive examples that satisfies: $a_1 \geq 1$, $b_1 \leq 1$ and for sampling negative instances we use the distribution with parameters: $a_0 \leq 1$, $b_0 \geq 1$. Practically it means, that we aim at putting the higher impact on the examples that are characterized by higher predictive probabil-

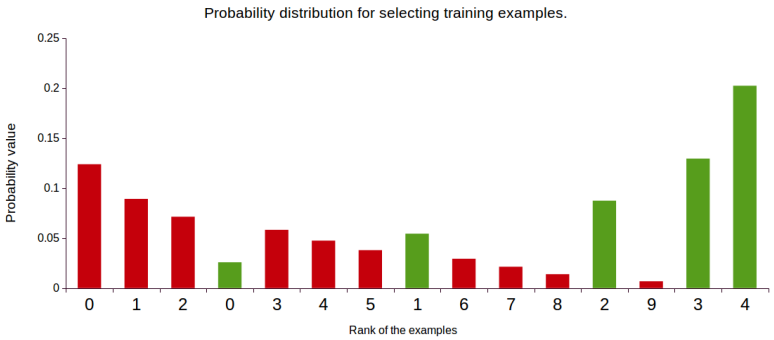
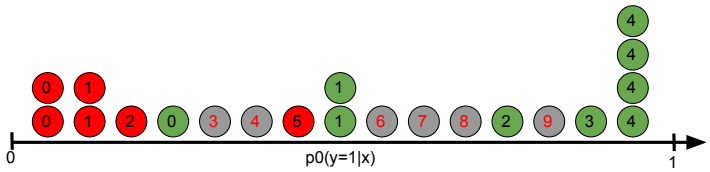
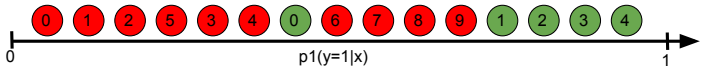


Fig. 3: Sampling distribution for examples presented in Figure 2 - $BBin_0(0.8,2,9)$ for negative and $BBin_1(2,0.8,4)$ for positive examples.



(a) The data sampled from distribution presented in Figure 3 (Grey circle stays behind unselected sample).



(b) The new order based on the classification of the ensemble model $p_1(y = 1|\mathbf{x}) = \frac{p(y=1|x,0)+p(y=1|x,1)}{2}$.

Fig. 4: The illustrative example presenting the capabilities of the joined ensemble model, after training the second base learner on the sampled data.

ities (for positive examples), or lower probability values (for negative examples). Our philosophy for this particular case is to put the higher impact on distinguishing the examples located away from each other in the global ranking determined by the predictions comparing to examples located in the weighted middle of the ranking list. As a consequence, we are sacrificing some portion of difficult to distinguish examples by putting them to unsure region, but we avoid observing them in low or high ranking positions. So the model has some capability to prevent overfitting that can be caused by discursive (or even noise) examples in training data. We also aim at dealing with imbalanced data phenomenon by sampling equal number of positive and negative examples.

Quite opposite strategy is observed for the following sampling distribution:

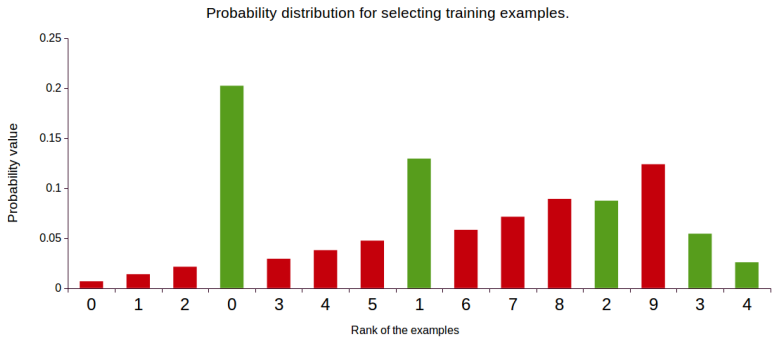


Fig. 5: Sampling distribution for examples presented in Figure 2 - $BBin_0(2, 0.8, 9)$ for negative and $BBin_1(0.8, 2, 4)$ for positive examples.

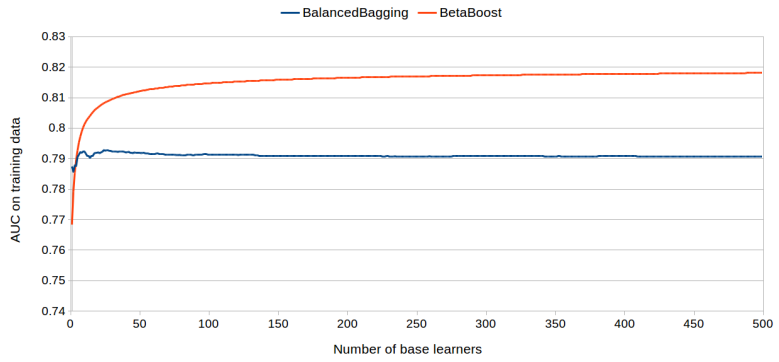
- $BBin_0(2, 0.8, 9)$ for negative examples.
- $BBin_1(0.8, 2, 4)$ for positive examples.

In this case, the sampling distribution for the next boosting iteration is presented in Figure 5. Following this strategy we aim at correct classification of the improperly ranked examples, assuming that they are rather hard examples that we manage to classify by the ensemble model.

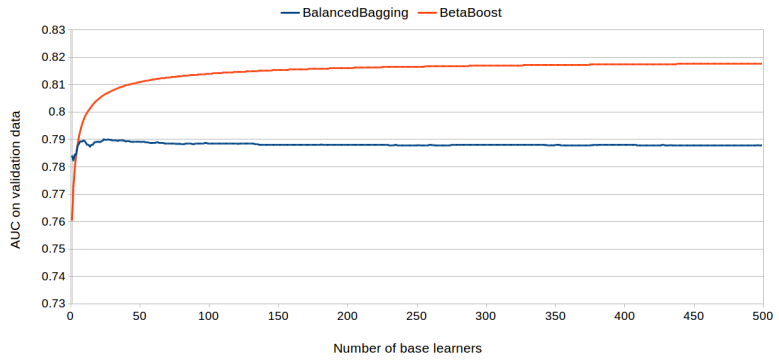
The two presented strategies aim at different cases. In the first case we trust our base model, but we do not trust to our data assuming that there are some portion of the examples that are impossible to be distinguished. Therefore, we are leaving some portion of examples in controversial area on the ranking, cleaning low and high ranking regions with improperly located samples. For the second strategy, we use rather untrusted weak learner as a base model, but we aim at create the complex model that will properly classify hard instances if their impact is going to be decreased.

2.4 Relation to existing solutions

The presented work is inspired by existing *RankBoost* [5] (for which the equivalence to well known *AdaBoost* was described in [19]) method and couple of other approaches. In contrast to the *RankBoost* we define two separate ranking functions for positive and negative examples. First of all, the *Rankboost* approach is very sensitive to the noisy examples located in training data. *BetaBoost* model presented in this paper deals well with insecure and noisy data because the distribution is not updated in iterations and does not depend on global ranking. Moreover, it is also more beneficial to use more flexible sampling distribution that is characterized by two parameters (a and b) contrary to the specific exponential-based distribution used



(a) Training set.



(b) Validation set.

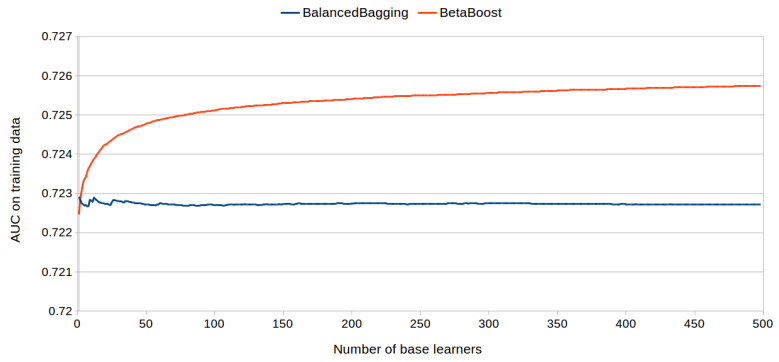
Fig. 6: A comparison analysis of *BetaBoost* ($a_0 = 0.8$, $b_0 = 2$, $a_1 = 2$ and $b_1 = 0.8$) and *Balanced Bagging* for the growing number of base learners on *GMSC* dataset. We consider *Logistic Regression* as base learner. *AUC* is taken as quality criterion.

in typical boosting approaches. The proposed solution is also inheriting self-paced philosophy [12] if the strategy with the increasing probabilities for positive and with decreasing probabilities for negative examples is applied.

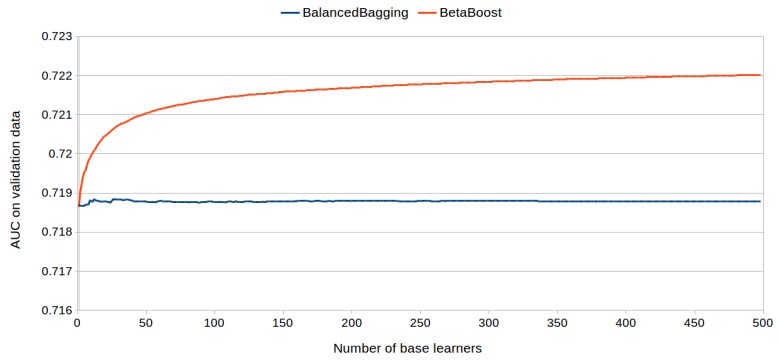
As the procedure is independent on global ranking it is crucial to apply proper model selection procedure that will fit proper sampling curves for each of the classes.

3 Experiments

We are going to evaluate our approach on two large datasets from credit scoring domain that are available in *Kaggle* repository:



(a) Training set.



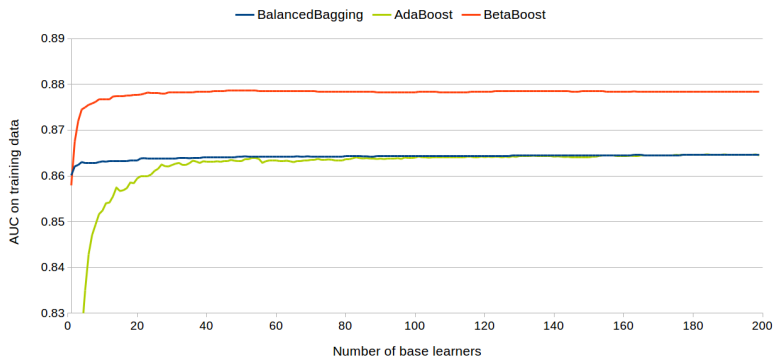
(b) Validation set.

Fig. 7: A comparison analysis of *BetaBoost* ($a_0 = 0.8$, $b_0 = 2$, $a_1 = 2$ and $b_1 = 0.8$) and *Balanced Bagging* for the growing number of base learners on *LCLD* dataset. We consider *Logistic Regression* as base learner. *AUC* is taken as quality criterion.

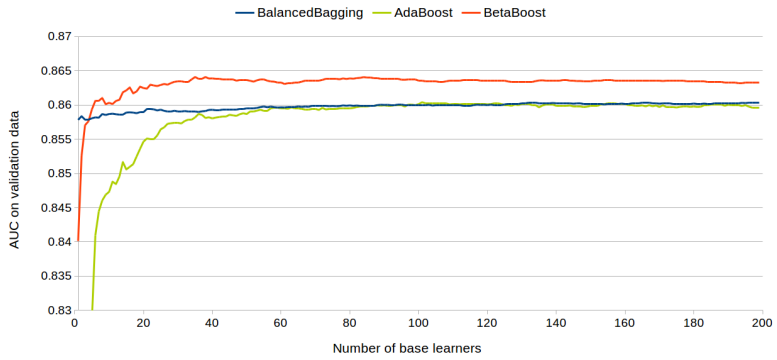
- *Give me Some Credit* [6].
- *Lending Club Loan Data* [14].

Give me Some Credit (GMSC) dataset is composed of 150000 examples, 10026 positive and 139974 negative elements. Each of the credit consumers is represented by the vector of 10 numeric features. Each of the attributes were normalized before using it for training.

Lending Club Loan Data(LCLD) dataset is composed of 887379 examples, 67429 positive and 819955 negative cases. Each of the examples were described by 12 features, where 6 of them were numeric, and the remaining 6 were nominal. On the preprocessing stage we have normalized the numeric features and binarized nominal attributes.



(a) Training set.



(b) Validation set.

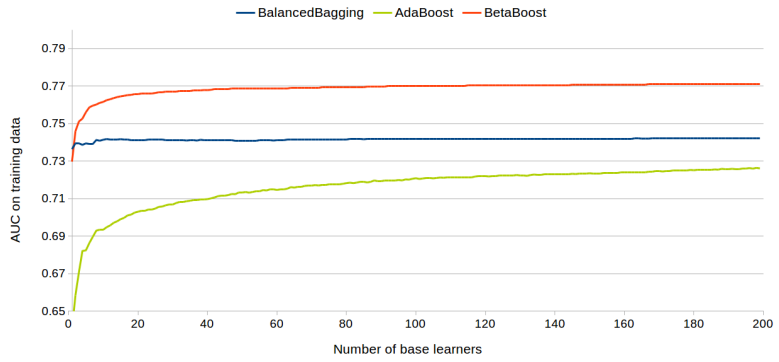
Fig. 8: A comparison analysis of *BetaBoost* ($a_0 = 1.5$, $b_0 = 0.8$, $a_1 = 0.8$ and $b_1 = 1.5$) and *Balanced Bagging* for the growing number of base learners on *GMSC* dataset. We consider *Logistic Regression* as base learner. *AUC* is taken as quality criterion.

We divide each of the initial datasets to: training set (80% examples) and test set (20% examples). From training set we separate 10% instances for validation to monitor the training progress and select the best set of base learners.

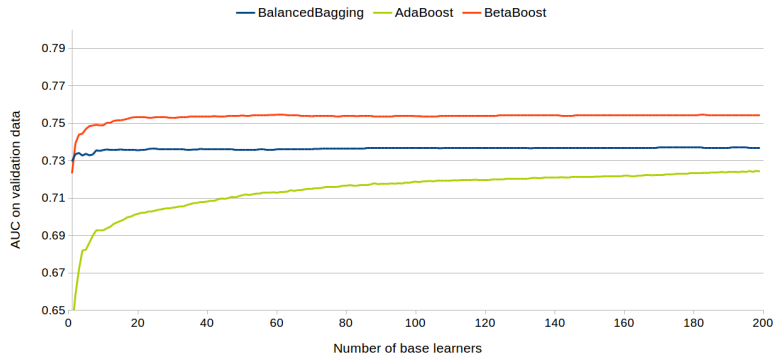
For the evaluation we use *AUC (area under ROC curve)* criterion, which is often for evaluating credit scoring models and measures well the sorting capabilities of learners. For each of the scenarios we apply model selection of the sampling parameters (a_1, b_1, a_0, b_0) from the set of candidates and select the parameters with the highest *AUC* obtained on the validation set.

We consider the two scenarios that were described in this work. In the first of the scenarios we aim at putting higher weights to the "secure" examples, assuming that controversial examples are hard to classify.

Therefore we propose to use *Logistic Regression* as a stable base learner:



(a) Training set.



(b) Validation set.

Fig. 9: A comparison analysis of *BetaBoost* ($a_0 = 1.2$, $b_0 = 0.8$, $a_1 = 0.8$ and $b_1 = 1.2$),*Balanced Bagging* and *AdaBoost* for the growing number of base learners on *LCLD* dataset. We consider *Decision Tree* as base learner. *AUC* is taken as quality criterion.

$$p(y = 1|x,k) = \sigma(\mathbf{w}_k^{\mathrm{T}}\mathbf{x}) = \frac{1}{1 - \exp\{-\mathbf{w}_k^{\mathrm{T}}\mathbf{x}\}} \tag{5}$$

At first we analyze the training capabilities of the *BetaBoost* model trained using the following beta parameters: $a_0 = 0.8$, $b_0 = 2$, $a_1 = 2$ and $b_1 = 0.8$. We compare the proposed approach with so called *Balanced Bagging* that performs sampling with replacement from uniform distribution to obtain $N/2$ samples from each class. The results of the comparison are presented in Figure 6 and 7.

It can be observed, that *Logistic Regression* is a very stable model characterized by small variance of the performance. Practically, it means that small changes in data caused by uniform sampling does not affect the overall performance of the

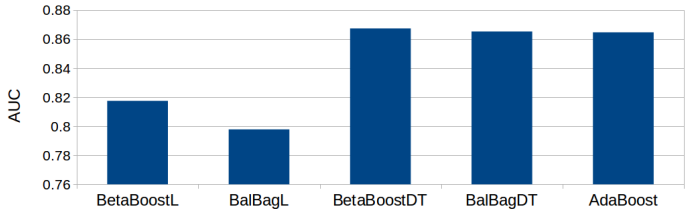


Fig. 10: Final results for considered models - *GMSC* dataset (test data)

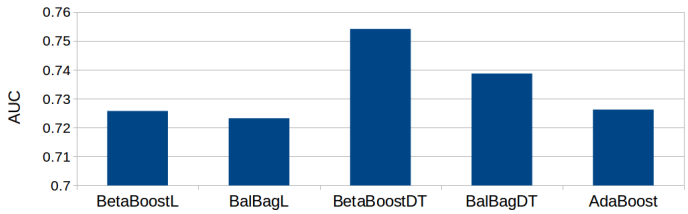


Fig. 11: Final results for considered models - *LCLD* dataset (test data)

model. If we apply sampling for the procedure characteristic for *BetaBoost* model we would obtain the improvement of *AUC* measure as it is observed in Figures 6 and 7. As a consequence of increasing probabilities for positive examples ($a_1 > 1$ and $b_1 < 1$) and decreasing probabilities for negative cases ($a_0 < 1$ and $b_0 > 1$) we aim at good quality prediction of the positive examples that are located on higher ranking positions and negative examples that are located on low positions. To obtain the goal we sacrifice the "difficult" examples that are suspected to be "noisy" instances, that are located in the discussion area. As a consequence, the improvement of *AUC* is observed for both of the considered datasets.

As a second base model we propose to use *Decision Trees*. As a fact that this model is recognized as so called *weak learner*, we propose the following sampling parameters to train the *BetaBoost* models:

- $a_0 = 1.5$, $b_0 = 0.8$, $a_1 = 0.8$ and $b_1 = 1.5$ for *GMSC* dataset,
- $a_0 = 1.2$, $b_0 = 0.8$, $a_1 = 0.8$ and $b_1 = 1.2$ for *LCLD* dataset.

The results are presented in Figures 7 and 8. We can see that sampling with replacement using the second strategy ($a_0 \geq 1$, $b_0 \leq 1$ and $a_1 \leq 1$, $b_1 \geq 1$) makes significant improvement of *AUC* criterion comparing to *BetaBoost* strategy, that also uses decision tree as a base learner. We also consider in the analysis the *AdaBoost* classifier that learns the component base model using the similar strategy that increases the impact of "hard examples", decreasing the significance of well predicted instances. The *AdaBoost* model needs more iterations to achieve acceptable *AUC* level because both datasets are spoiled by imbalanced data phenomenon. The performance of *AdaBoost* is similar to *BetaBoost* on *GMSC* dataset, but on *LCLD* dataset

it gives significantly worse results. We also present the results on validation data to show that overfitting problem is not observed for the considered models.

We presented the final results obtained by the considered models in Figures 10 (*GMSC* dataset) and 11 *LCLD* dataset. The considered models are as follows:

- **BetaBoostL**. *BetaBoost* with *Logistic Regression* as base learner trained with the first strategy ($a_0 \leq 1$, $b_0 \geq 1$ and $a_1 \geq 1$, $b_1 \leq 1$).
- **BalBagL**. *Balanced Bagging* with *Logistic Regression* as a base learner.
- **BetaBoostDT**. *BetaBoost* with a decision tree as a base learner trained with the second strategy ($a_0 \geq 1$, $b_0 \leq 1$ and $a_1 \leq 1$, $b_1 \geq 1$).
- **BalBagDT**. *Balanced Bagging* with a decision tree as a base learner.
- **AdaBoost**. *AdaBoost* classifier with a decision tree as a base learner.

It can be observed, that the *BetaBoost* with decision tree as a base learner train with the second strategy performed better then the reference approaches considered in the experiments. On (*GMSC* dataset) we observed only slight increase in quality of *BetaBoost* comparing to *Balanced Bagging* from 0.8652 to 0.8673. However, we operate on *big data*, so the slight improvements in quality criterion may have great impact on financial benefit. The improvement observed on the *LCLD* dataset is indisputable.

4 Conclusions and Future Works

In this work we propose alternative ensemble based strategy, that makes use of beta binomial sampling to create the base models. Two strategies can be distinguished while taking the sampling distribution. In the first strategy we aim at putting higher impact on "easy examples", we bestow trust the base model and do not trust in data quality. In the second strategy we take rather weak and unstable base model and we put the higher impact on training "hard examples".

Contrary to existing approaches like *AdaBoost*, we update the sampling distribution basing only on individual ranking for each of the classes. As a consequence, the impact of noisy examples in training data is not high.

The crucial step for the proposed the *BetaBoost* model is to find proper parameters for sampling distributions. It can be performed by grid search, but this approach is ineffective for large data sets. In the future works we plan to propose the smart model selection approach to solve that issue. Additionally, we are going to perform more formal discussion of the properties of the proposed model. Moreover, the weighted variant of ensemble model is going to be proposed.

References

1. Abellán J, Mantas CJ (2014) Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring. Expert Systems with Applications 41(8):3825–3830
2. Bellotti T, Crook J (2009) Support Vector Machines for credit scoring and discovery of significant features. Expert Systems with Applications 36(2):3302–3308
3. Chen S, Härdle WK, Jeong K (2010) Forecasting volatility with Support Vector Machine-based GARCH model. Journal of Forecasting 29(4):406–433
4. Chen S, Härdle W, Moro R (2011) Modeling default risk with Support Vector Machines. Quantitative Finance 11(1):135–154
5. Freund Y, Iyer R, Schapire RE, Singer Y (2003) An efficient boosting algorithm for combining preferences. Journal of machine learning research 4(Nov):933–969
6. Give me Some Credit (2011) Give me Some Credit. https://www.kaggle.com/c/GiveMeSomeCredit
7. Härdle W, Lee YJ, Schäfer D, Yeh YR (2009) Variable selection and oversampling in the use of smooth Support Vector Machines for predicting the default risk of companies. Journal of Forecasting 28(6):512–534
8. Härdle WK, Prastyo DD, Hafner C (2012) Support Vector Machines with Evolutionary Feature Selection for Default Prediction. Handbook of Applied Nonparametric and Semi-parametric Econometrics and Statistics pp 346–373
9. Harris T (2015) Credit scoring using the clustered Support Vector Machine. Expert Systems with Applications 42(2):741–750
10. Huang SC (2011) Using Gaussian process based kernel classifiers for credit rating forecasting. Expert Systems with Applications 38(7):8607–8611
11. Koutanaei FN, Sajedi H, Khanbabaei M (2015) A hybrid data mining model of feature selection algorithms and ensemble learning classifiers for credit scoring. Journal of Retailing and Consumer Services 27:11–23
12. Kumar MP, Packer B, Koller D (2010) Self-paced learning for latent variable models. In: Advances in Neural Information Processing Systems, pp 1189–1197
13. Lee TS, Chiu CC, Lu CJ, Chen IF (2002) Credit scoring using the hybrid neural discriminant technique. Expert Systems with applications 23(3):245–254
14. Lending Club (2016) Lending Club Loan Data. https://www.kaggle.com/wendykan/lending-club-loan-data
15. Marqués A, García V, Sánchez JS (2012) Two-level classifier ensembles for credit risk assessment. Expert Systems with Applications 39(12):10,916–10,922
16. Martens D, Baesens B, Van Gestel T, Vanthienen J (2007) Comprehensible credit scoring models using rule extraction from Support Vector Machines. European journal of operational research 183(3):1466–1476

17. Nanni L, Lumini A (2009) An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring. Expert systems with applications 36(2):3028–3033
18. Oreski S, Oreski D, Oreski G (2012) Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment. Expert systems with applications 39(16):12,605–12,617
19. Rudin C, Schapire RE (2009) Margin-based ranking and an equivalence between AdaBoost and RankBoost. Journal of Machine Learning Research 10(Oct):2193–2232
20. Tomczak JM, Zięba M (2015) Classification Restricted Boltzmann Machine for comprehensible credit scoring model. Expert Systems with Applications 42(4):1789–1796
21. Tsai CF, Wu JW (2008) Using neural network ensembles for bankruptcy prediction and credit scoring. Expert systems with applications 34(4):2639–2649
22. Zhao Z, Xu S, Kang BH, Kabir MMJ, Liu Y, Wasinger R (2015) Investigation and improvement of multi-layer perceptron neural networks for credit scoring. Expert Systems with Applications 42(7):3508–3516
23. Zhou L, Lai KK, Yen J (2009) Credit scoring models with AUC maximization based on weighted SVM. International journal of information technology & decision making 8(04):677–696
24. Zhu Y, Xie C, Wang GJ, Yan XG (2016) Comparison of individual, ensemble and integrated ensemble machine learning methods to predict chinas sme credit risk in supply chain finance. Neural Computing and Applications pp 1–10
25. Zięba M, Świątek J (2012) Ensemble classifier for solving credit scoring problems. In: Doctoral Conference on Computing, Electrical and Industrial Systems, Springer, pp 59–66
26. Zięba M, Tomczak JM (2015) Boosted svm with active learning strategy for imbalanced data. Soft Computing 19(12):3357–3368
27. Zięba M, Tomczak SK, Tomczak JM (2016) Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction. Expert Systems with Applications 58:93–101

SFB 649 Discussion Paper Series 2016

For a complete list of Discussion Papers published by the SFB 649, please visit http://sfb649.wiwi.hu-berlin.de.

001 "Downside risk and stock returns: An empirical analysis of the long-run and short-run dynamics from the G-7 Countries" by Cathy Yi-Hsuan Chen, Thomas C. Chiang and Wolfgang Karl Härdle, January 2016.
002 "Uncertainty and Employment Dynamics in the Euro Area and the US" by Aleksei Netsunajev and Katharina Glass, January 2016.
003 "College Admissions with Entrance Exams: Centralized versus Decentralized" by Isa E. Hafalir, Rustamdjan Hakimov, Dorothea Kübler and Morimitsu Kurino, January 2016.
004 "Leveraged ETF options implied volatility paradox: a statistical study" by Wolfgang Karl Härdle, Sergey Nasekin and Zhiwu Hong, February 2016.
005 "The German Labor Market Miracle, 2003 -2015: An Assessment" by Michael C. Burda, February 2016.
006 "What Derives the Bond Portfolio Value-at-Risk: Information Roles of Macroeconomic and Financial Stress Factors" by Anthony H. Tu and Cathy Yi-Hsuan Chen, February 2016.
007 "Budget-neutral fiscal rules targeting inflation differentials" by Maren Brede, February 2016.
008 "Measuring the benefit from reducing income inequality in terms of GDP" by Simon Voigts, February 2016.
009 "Solving DSGE Portfolio Choice Models with Asymmetric Countries" by Grzegorz R. Dlugoszek, February 2016.
010 "No Role for the Hartz Reforms? Demand and Supply Factors in the German Labor Market, 1993-2014" by Michael C. Burda and Stefanie Seele, February 2016.
011 "Cognitive Load Increases Risk Aversion" by Holger Gerhardt, Guido P. Biele, Hauke R. Heekeren, and Harald Uhlig, March 2016.
012 "Neighborhood Effects in Wind Farm Performance: An Econometric Approach" by Matthias Ritter, Simone Pieralli and Martin Odening, March 2016.
013 "The importance of time-varying parameters in new Keynesian models with zero lower bound" by Julien Albertini and Hong Lan, March 2016.
014 "Aggregate Employment, Job Polarization and Inequalities: A Transatlantic Perspective" by Julien Albertini and Jean Olivier Hairault, March 2016.
015 "The Anchoring of Inflation Expectations in the Short and in the Long Run" by Dieter Nautz, Aleksei Netsunajev and Till Strohsal, March 2016.
016 "Irrational Exuberance and Herding in Financial Markets" by Christopher Boortz, March 2016.
017 "Calculating Joint Confidence Bands for Impulse Response Functions using Highest Density Regions" by Helmut Lütkepohl, Anna Staszewska-Bystrova and Peter Winker, March 2016.
018 "Factorisable Sparse Tail Event Curves with Expectiles" by Wolfgang K. Härdle, Chen Huang and Shih-Kang Chao, March 2016.
019 "International dynamics of inflation expectations" by Aleksei Netšunajev and Lars Winkelmann, May 2016.
020 "Academic Ranking Scales in Economics: Prediction and Imdputation" by Alona Zharova, Andrija Mihoci and Wolfgang Karl Härdle, May 2016.

SFB 649, Spandauer Straße 1, D-10178 Berlin
http://sfb649.wiwi.hu-berlin.de

This research was supported by the Deutsche Forschungsgemeinschaft through the SFB 649 "Economic Risk".

SFB 649 Discussion Paper Series 2016

For a complete list of Discussion Papers published by the SFB 649, please visit http://sfb649.wiwi.hu-berlin.de.

021 "CRIX or evaluating blockchain based currencies" by Simon Trimborn and Wolfgang Karl Härdle, May 2016.
022 "Towards a national indicator for urban green space provision and environmental inequalities in Germany: Method and findings" by Henry Wüstemann, Dennis Kalisch, June 2016.
023 "A Mortality Model for Multi-populations: A Semi-Parametric Approach" by Lei Fang, Wolfgang K. Härdle and Juhyun Park, June 2016.
024 "Simultaneous Inference for the Partially Linear Model with a Multivariate Unknown Function when the Covariates are Measured with Errors" by Kun Ho Kim, Shih-Kang Chao and Wolfgang K. Härdle, August 2016.
025 "Forecasting Limit Order Book Liquidity Supply-Demand Curves with Functional AutoRegressive Dynamics" by Ying Chen, Wee Song Chua and Wolfgang K. Härdle, August 2016.
026 "VAT multipliers and pass-through dynamics" by Simon Voigts, August 2016.
027 "Can a Bonus Overcome Moral Hazard? An Experiment on Voluntary Payments, Competition, and Reputation in Markets for Expert Services" by Vera Angelova and Tobias Regner, August 2016.
028 "Relative Performance of Liability Rules: Experimental Evidence" by Vera Angelova, Giuseppe Attanasi, Yolande Hiriart, August 2016.
029 "What renders financial advisors less treacherous? On commissions and reciprocity" by Vera Angelova, August 2016.
030 "Do voluntary payments to advisors improve the quality of financial advice? An experimental sender-receiver game" by Vera Angelova and Tobias Regner, August 2016.
031 "A first econometric analysis of the CRIX family" by Shi Chen, Cathy Yi-Hsuan Chen, Wolfgang Karl Härdle, TM Lee and Bobby Ong, August 2016.
032 "Specification Testing in Nonparametric Instrumental Quantile Regression" by Christoph Breunig, August 2016.
033 "Functional Principal Component Analysis for Derivatives of Multivariate Curves" by Maria Grith, Wolfgang K. Härdle, Alois Kneip and Heiko Wagner, August 2016.
034 "Blooming Landscapes in the West? - German reunification and the price of land." by Raphael Schoettler and Nikolaus Wolf, September 2016.
035 "Time-Adaptive Probabilistic Forecasts of Electricity Spot Prices with Application to Risk Management." by Brenda López Cabrera , Franziska Schulz, September 2016.
036 "Protecting Unsophisticated Applicants in School Choice through Information Disclosure" by Christian Basteck and Marco Mantovani, September 2016.
037 "Cognitive Ability and Games of School Choice" by Christian Basteck and Marco Mantovani, Oktober 2016.
038 "The Cross-Section of Crypto-Currencies as Financial Assets: An Overview" by Hermann Elendner, Simon Trimborn, Bobby Ong and Teik Ming Lee, Oktober 2016.
039 "Disinflation and the Phillips Curve: Israel 1986-2015" by Rafi Melnick and Till Strohsal, Oktober 2016.

SFB 649, Spandauer Straße 1, D-10178 Berlin
http://sfb649.wiwi.hu-berlin.de

This research was supported by the Deutsche Forschungsgemeinschaft through the SFB 649 "Economic Risk".

SFB 649 Discussion Paper Series 2016

For a complete list of Discussion Papers published by the SFB 649, please visit http://sfb649.wiwi.hu-berlin.de.

040 "Principal Component Analysis in an Asymmetric Norm" by Ngoc M. Tran, Petra Burdejová, Maria Osipenko and Wolfgang K. Härdle, October 2016.
041 "Forward Guidance under Disagreement - Evidence from the Fed's Dot Projections" by Gunda-Alexandra Detmers, October 2016.
042 "The Impact of a Negative Labor Demand Shock on Fertility - Evidence from the Fall of the Berlin Wall" by Hannah Liepmann, October 2016.
043 "Implications of Shadow Bank Regulation for Monetary Policy at the Zero Lower Bound" by Falk Mazelis, October 2016.
044 "Dynamic Contracting with Long-Term Consequences: Optimal CEO Compensation and Turnover" by Suvi Vasama, October 2016.
045 "Information Acquisition and Liquidity Dry-Ups" by Philipp Koenig and David Pothier, October 2016.
046 "Credit Rating Score Analysis" by Wolfgang Karl Härdle, Phoon Kok Fai and David Lee Kuo Chuen, November 2016.
047 "Time Varying Quantile Lasso" by Lenka Zbonakova, Wolfgang Karl Härdle, Phoon Kok Fai and Weining Wang, November 2016.
048 "Unraveling of Cooperation in Dynamic Collaboration" by Suvi Vasama, November 2016.
049 "Q3-D3-LSA" by Lukas Borke and Wolfgang K. Härdle, November 2016.
050 "Network Quantile Autoregression" by Xuening Zhu, Weining Wang, Hangsheng Wang and Wolfgang Karl Härdle, November 2016.
051 "Dynamic Topic Modelling for Cryptocurrency Community Forums" by Marco Linton, Ernie Gin Swee Teo, Elisabeth Bommes, Cathy Yi-Hsuan Chen and Wolfgang Karl Härdle, November 2016.
052 "Beta-boosted ensemble for big credit scoring data" by Maciej Zieba and Wolfgang Karl Härdle, November 2016.

SFB 649, Spandauer Straße 1, D-10178 Berlin
http://sfb649.wiwi.hu-berlin.de

This research was supported by the Deutsche Forschungsgemeinschaft through the SFB 649 "Economic Risk".