HUMBOLDT-UNIVERSITÄT ZU BERLIN


# Winning the Data Mining Cup competition 2015 – Challenges and success factors

Seminar paper in

**Applied Predictive Analytics**

Prof. Dr. Stefan Lessmann

Humboldt-Universität zu Berlin

School of Business and Economics

Chair of Information Systems

Alexander Dautel 561296

Martin Müller 539010

Berlin, September 11, 2015

# TABLE OF CONTENTS


1 Introduction ..... 3

1.1 Task ..... 3

1.2 Organization and workflow ..... 3

2 Data ..... 4

2.1 Dataset characteristics ..... 4

2.2 Data origin ..... 5

2.3 Data preparation and Feature Engineering ..... 5

3 Performance of Machine Learning algorithms ..... 6

4 Performance of econometric approaches ..... 8

5 Forecast combination ..... 9

5.1 Ensemble selection ..... 9

5.2 Simple averaging of best-performing base models ..... 9

5.3 Ensembles drawn from random classifier subspaces ..... 10

5.4 Performance of forecast combinations ..... 10

6 Conclusion ..... 12

7 Appendix ..... 13

# 1 INTRODUCTION

In the context of the seminar ”Applied Predictive Analytics” at Humboldt-Universität zu Berlin (HU Berlin) two student teams were able to secure places in the Data Mining Cup competition 2015. One team placed 6th, while the other team secured the 1st place of all participating teams to become the overall winner. This report is trying to provide a brief overview of the steps along the way and to examine some selected success factors.

## 1.1 TASK

The Data Mining Cup (DMC) is an annual competition between universities, organized by prudsys AG to get students excited for intelligent data analysis. In this year's edition, the competing teams' task was to investigate the influence of coupons on the order behavior of customers of an online store. From historical order data the following four target variables were to be predicted:

- coupon redemption probabilities for three different coupons (T1-T3)
- the absolute basket value of each transaction (BV)

For the evaluation of the students' forecasts, a loss function was constructed specially for this contest. This loss function punished the absolute deviations from the actual values, set in relation to the attribute mean, (quadratic) – a significantly modified version of the Mean Square Percentage Error – and summed over all data points for the four target variables. The participating teams were ranked in ascending order by the loss function value of their contribution. From publication of the task to the entry deadline, six weeks were given to work on the project. All participating students of HU Berlin worked together throughout the duration of the competition and formed two different teams purely for formal reasons.

## 1.2 ORGANIZATION AND WORKFLOW

To exploit the various academic backgrounds of the team members and account for the tight schedule, loose working groups were formed, which were each responsible for a particular aspect of the data analysis process:

- *Infrastructure and competition specifics*
- *Data pre-processing and feature engineering*
- *Application of Machine Learning algorithms*
- *Forecast combination*
- *Econometric insight*

The progress inside those subgroups was discussed with the whole team weekly. According to the needs and our progress in the competition, each team member's group allocation and responsibilities of changed over time.

After some time in the competition, it became clear that approaches other than classic machine learning algorithms did not yield any superior results, mainly due to

the high number of independent variables. This resulted in basically two areas people were working in, namely data pre-processing and the training and com of machine learning models. Just as about half the time had passed, we were able to figure out to which specific online store the dataset provided by prudsys belonged. This was helpful developing some alternative models, which we believe ultimately gave us some advantage predicting the basket values, as shown in later sections.

Regarding data preparation, due to the long time it took to train all the models on newly pre-processed data, we were only able to compare two versions during the competition period. In retrospect, we can show that the final version provided by the data team fortunately proofed to be superior to at least two previous versions. The decision to train a large number of machine-learning models on the available data was inspired by a 2009 paper authored by the winners of that year's KDD Cup. We see this as a very beneficial decision, as it emerged in the post competition that less common classifiers did reasonably well in predicting the unlabeled data. Finally, our approach to ensembling was to use the ensemble selection algorithm, which adds models to the ensemble based on the improvement in predictive power on a test data set. Surprisingly, this algorithm actually turned out to be inferior to simply combining the best single classifiers or drawing randomly from a sample of all classifiers with probabilities weighted by their individual performance on the test set.

# 2 DATA

## 2.1 DATASET CHARACTERISTICS

The data set contained 6 722 observations, of which 6 053 were fully labelled and 669 were missing the attribute values for the four target variables. This implied roughly a 9:1 ratio of total training to classification data. The fully labelled part of the data set was split into a training and a test data set to evaluate the various models and model combinations; those sets are henceforth referred to as the *training set* and *test set*, the data with missing target variable values as the *classification set*.

All observations were single transactions taken from an online shop and could be distinguished by a unique transaction identifier. In the original dataset each possessed another 31 attributes: the four target variables described above, two timestamp variables for the time of coupon reception and the time of coupon usage, as well as 24 other variables containing transaction specific information.

Through feature engineering by the data preparation subgroup, another 31 additional variables could be created and integrated into the dataset over the time of the competition so that the final dataset contained 63 variables. This process resulted in considerable improvements in the accuracy of the predictions.

## 2.2 DATA ORIGIN

An important insight into the nature of the data was worked out by the *econometric and alternative approaches* subgroup: plotting the order frequency of recurring customers, they detected a pattern of frequent 7 and 14 day intervals between purchases. A histogram of basket values suggested furthermore that customers preferably ordered goods for a total value just above a number of thresholds; when considering value added tax, these threshold values matched perfectly with common minimum order values for free or reduced shipping costs.

Together with the characteristics of small item prices, but large overall basket values this strongly indicated that the data originated from an online supermarket. By researching different tax rates and a list of the organizing company's customers, the team could determine the source of the data with great confidence. This helped tremendously in understanding the data and in making decisions where pure computing power did not seem to improve our results. The median-based approach to predicting basket values for known users (further described in a separate report by the *Econometric insight and alternative approaches* subgroup) hails directly from this insight.

## 2.3 DATA PREPARATION AND FEATURE ENGINEERING

The *Data pre-processing and feature engineering* subgroup operated in several working steps, continually updating the datasets available to the other subgroups. Data pre-processing proved to be a minor task due to very few missing values and very low prioritization of outlier treatment in early stages.

Using numerous proven techniques combined with human insight, a considerable number of new variables could be created from the data set: First, the two existing timestamp variables were used to create additional time and factor variables like time of the day, weekday or duration between purchases for a given customer. In the next step, a the operating principle of recommendation engines was applied to product groups and brand that customers in the dataset were shopping; in this way, customers could be clustered by their shopping behavior. Subsequently, the weight of evidence method was utilized to transform large factor variables and thus make them available to methods that do not handle factor variables well. In a last step, base rates (mainly mean and median) for observations with equal factor levels were computed and served as additional information for the learning algorithms.

For the post-competitions analysis, predictions from the initial dataset (*1.0*) were compared with an intermediate (*3.2*) and the final dataset (*6.3*) found in table 3.2.

| Data version | Number of variables |
|---|---|
| 1.0 | 10 |
| 3.2 | 37 |
| 5.2 | 64 |
| 6.3 | 63 |

Table 3.1: Selected data versions

A more detailed description of the whole feature engineering process can be found in the *Data pre-processing and feature engineering* subgroup's separate report.

# 3 PERFORMANCE OF MACHINE LEARNING ALGORITHMS

For the actual process of predicting the target variables we took the same approach suggested in a 2009 paper composed by the winners of this year's KDD Cup. That is, training as many algorithms as possible on the available data in order to gain a picture of which perform best on the given dataset and also to have a wide range of algorithms available for the ensembling process.

The whole procedure of building models based on the training data and predicting cases in the test data set was again realized in the statistical programming language R. We primarily made use of the algorithms available in the caret package to achieve this, since it allows for a convenient way to create a large number of models. The complete list of algorithms can be found in tables 4.1-4.4. They show the performance of each model for differently pre-processed versions of the data and the coupon or basket value targets, respectively. Performance in this context means the score of the loss function for predicting the values of the previously unlabeled data, the values of which we only received after the competition ended. The score is easily obtained by comparing model predictions with the actual values.

Even though the *Data pre-processing and feature engineering* team produced several data sets, varying in number of variables and structure, we were only able train models on the very first version (*1.0*) and the final version (*6.3*) during the competition period. Main reason for this was the speed with which the data group produced new data sets and the amount of time it took to train all models on newly available data, up to 24 hours. We therefore built the setup necessary for the training and testing process employing data set 1.0 and then set a deadline for the data team to submit their final version several days before the deadline of the competition. In order to get a picture of how intermediate versions of the data would have performed, we repeated the process for version 3.2 in the post-competition analysis. The table also includes some missing values. The *econ* model was provided by the group responsible for econometric approaches to forecasting for the final data version only. Additionally, the setup for predicting basket values was established relatively late in the competition, so basket value predictions were only available for versions 5.2 and

6.3. During the process of training and evaluating models based on the intermediate data version 3.2, we encountered problems when employing algorithms based on support vector machines, i.e. we experienced software failure every time in the training process, and could not get a score for the coupons T1 and T2 for any of the models.

Table 4.4 also includes a column that shows the rank of each model based on how well it predicted the test data, which was basically all information available to us before submission of our predictions. Comparing the total score of the loss function would not be of any value here, since this metric depends on the number of observations, which is not equal for the test set and classification set. One can see that models which performed well on the test set usually also fit well on the classification set, with a few notable exceptions. This might be an indicator that overfitting on the test data was probably avoided. Still, especially in the lower ranks, models that previously performed relatively well do worse on the class data and vice versa.

Regarding the coupon redemption prediction, some interesting facts emerged in the final analysis. Firstly, the so-called *econ* model provides an even better performance than in the testing stage. Especially in predicting probabilities for coupon T2 this model outperforms other, "off-the-shelf" algorithms by far and jumps to first place among all base models with more than 300 points ahead of the second model. It also predicts coupon T1 redemption probabilities better than expected in the previous testing on the test data, as shown by the large improvement from eleventh to first place. Thus, we consider the deployment of this model one of the decisive factors to win the DMC 2015. In the following section, the development and characteristics of this particular model will be further outlined.

Rankings and scores of an individual classifier depend a lot on data set and target variable. As table 4.3 in the appendix shows, scores of the top ten classifiers differ by not more than 50 points for the initial version of the data set (*1.0*). In version *3.2* however, accuracy of the best models does not improve by a large margin, but some deliver a much worse prediction. Regarding version *6.3*, things look again similar to the situation in the basic data set.

For basket value estimation, a partly different set of classifiers was used, since the prediction of continuous variables automatically excludes certain algorithms. Again, the relatively simple models provided by the econometrics group dominate most commonly used machine-learning algorithms. It will be further elaborated on these in the following section. Surprisingly, the *boosted glm* and *svm with polynomial kernel* models jump from the very last ranks in the testing stage to rank two and three in the prediction on the classification set. The considerable estimation of a single outlier in the classification set adds to the success of these two models, as shown in the following section. Random forests uniformly deliver worse results than the top five classifiers, but for other families, such as support vector machines, the picture is more heterogeneous. Additionally, the data manipulation that took place between version *3.2* and *6.3* had a positive impact on every classifier's performance.

# 4 PERFORMANCE OF ECONOMETRIC APPROACHES

The approach taken by the econometrics group to predict coupon redemption probabilities was basically a carefully tuned random forest (the group name might be misleading here, but we figured out that econometric approaches were not working successfully early in the competition). This is even more surprising since standard random forest classifiers performed a lot weaker, as shown in table 4.1. The difference was mainly due to the tuning process. While the *Machine Learning* group primarily trained the standard models using a small number of predefined parameters from the *caret* package and let the software select the optimal ones based on the performance in the training process, the *Econometric insight* group experimented with a large range of different *mtry* values (number of variables randomly sampled as candidates at each split). They ended up with a very low value of 2, which is way below the standard value of the square root of the number of variables used in the training process. They also show that results on the classification set could have been further improved by reducing the *ntree* parameter (number of trees in the forest) and number of variables included.

Regarding the basket value predictions, the deployed approach was based on the insight that the data was coming from an online grocery store. Assuming that consumer needs for basic non-durable goods and food do not change much over a short time period, the model simply took the median value of previous purchases per customer as estimator for the basket value of the order that was to be predicted. For first-time purchasers, the value was estimated using a standard random forest model. Our team members were able to show in their analysis, that the success of this rather simplistic model was due to mainly one outlier, an order in the classification set with a value of about 11 000 (median value was 212). This user had made two previous purchases with basket values of around 4 000 and 6 000, so the median model estimated his third order to be worth just a little below the value of 5 000 and was by far the best single classifier in predicting this observation. The *glmboost* and *svmpoly* models also did reasonably well, predicting a value of roughly 4 000, still way ahead of the other classifiers, which all estimated the value to be below 1 000. Even with the best estimate the contribution of this single observation to the overall score of the basket value prediction was more than a third. This shows how vital this prediction was for the overall success in the competition. Ironically, the model excluded estimations above 5 000 and replaced them by random forest estimates, just as for new customers. Had the previous two high value orders been just 100 currency units higher, our team would probably not have won the Data Mining Cup competition 2015.

In their post-competition analysis, the Econometric insight group repeated the prediction on the classification set excluding this outlier and showed that the median based approach now is inferior to all other employed classifiers, which seem to be able to extract more information from the training data than this simplistic approach. One could now believe that this outlier in basket value is responsible for our teams'

success, but an exchange with other teams after the end of the competition revealed that many of them also used some sort of median based approach, as it performed reasonably well on the test data.

# 5 FORECAST COMBINATION

A number of different arguments can be found for using the combination of single forecasts (ensembling) instead of the predictions of individual models. By constructing clever ensembles of individual forecasts, in many cases one can reduce the bias and the variance of a prediction. Furthermore, the track record of forecast combinations in various data mining competitions speaks for itself: almost all winning solutions we could find used some kind of ensembling method.

## 5.1 ENSEMBLE SELECTION

From the university lectures *Business Analytics and Predictive Modelling* at HU Berlin some team members were familiar with a number of different combination strategies. A strategy called *ensemble selection* seemed well-suited for the underlying task. Ensemble selection is able to combine multiple base models stemming from different classification algorithms. It does so in a multi-step approach, by building the best system consisting of one base model (i.e. the single best individual model) and then iteratively adding the base models that improve the ensembles performance most until no such enhancement is possible.

The basic ensemble selection algorithm can be refined in several ways. Notably, the *forecast combination subgroup* set out to implement an ensemble selection strategy similar to that used in the KDD 2009 winning paper.

In theory and from other groups' experience in historic data analysis competitions, this forecast combination strategy should have produced near optimal results. However, in this particular competition, while ensemble selection improved our predictions compared against most individual models, we had more success by applying other combination techniques. Further analysis of the implemented ensemble selection algorithm can be found in a separate report by students from the *forecast combination subgroup*.

## 5.2 SIMPLE AVERAGING OF BEST-PERFORMING BASE MODELS

A very simple quick fix to the forecast combination problem is to compute the average of a given number of individual forecasts. This alone ensures different "views" on the data and has the potential to reduce bias and variance of the prediction. In our setting, typically the combination of the best two to ten individual models proved to perform best; especially for those target variables where the difference in accuracy between the best base models was small (mainly the coupon redemption probabilities) , this method lead to substantial improvements.

## 5.3 ENSEMBLES DRAWN FROM RANDOM CLASSIFIER SUBSPACES

In an effort to find some even better ensembles than those obtained through ensemble selection or simple averaging, the team designed their own algorithm similar to the *random subspace method*. The goal was to allow for more randomness than in the previous approaches and thus to detect any potentially promising forecast combinations that might have been overlooked.

From the space of all individual base models, a set of classifiers was drawn with replacement (thus allowing multiple occurrences of one classifier in the set). The drawing probabilities for the classifiers were inverse to their individual performance on the test set. The predictions of all classifiers in the set were averaged to obtain the ensemble prediction.

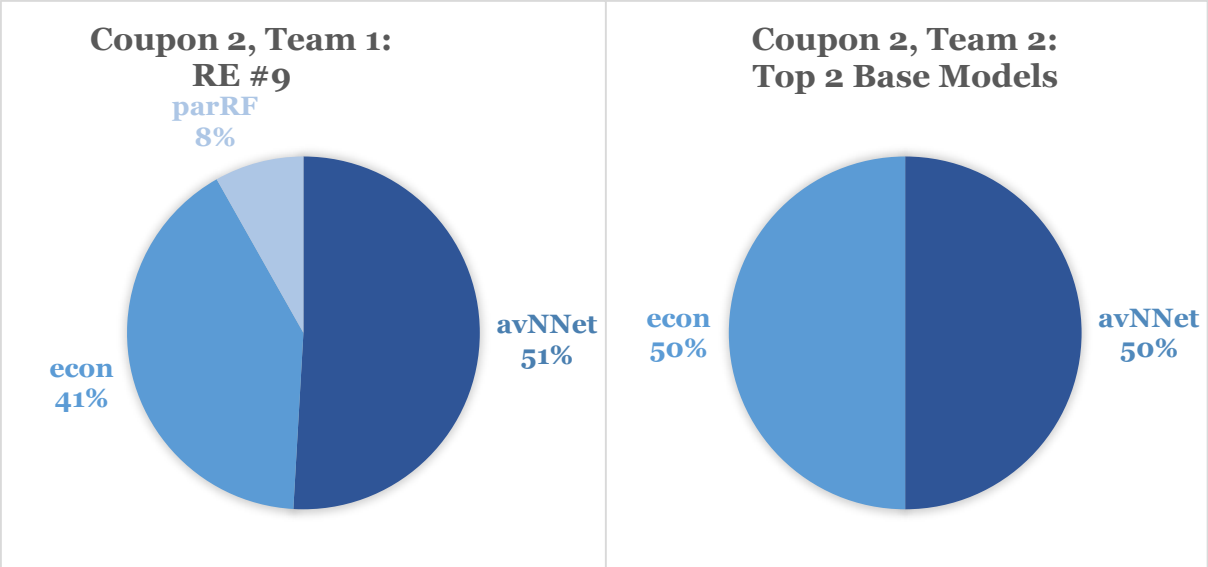
While the resulting ensemble may be similar in composition to those obtained through ensemble selection – basically a weighted average of base model predictions – this method did not exclude ensembles starting with "bad" individual prediction accuracy and generally operated in a more unstructured way.

Clearly, this algorithm is not a very (time and computing power) efficient solution, but due to the relatively small data set and number of base models, this did not pose any difficulties in the task at hand.

## 5.4 PERFORMANCE OF FORECAST COMBINATIONS

An analysis of the best performing forecast combinations revealed interesting insights: Table 5.1 in the appendix allows an overview over some of the ensembles that performed best on the classification set and their respective ranks on the test set. The submitted ensembles are highlighted; the best-ranked ensemble on the test set was submitted by Team 1 for all target variables.

The random subspace ensembles (RE) #18 and #17 for coupon 1 contained predictions from the *econ* forecast, which did not rank in the top 10 individual forecasts (see table 4.1) and was thus not considered by the ensemble selection or simple average approaches. The success of these ensembles for coupon 1 on both test *and* classification set is a perfect example for how one ostensibly bad model can boost an ensembles performance. That this is not always the case is shown by the ranking of ensembles for coupons 2 and 3: For both coupons, the simple average of a few best individual models worked best on the classification set, although not ranking highest in our test set comparisons.




**Graphic 5.1**: Ensemble composition for coupon 2 by submission

Graphic 5.1 shows how a seemingly small change in composition can lead to relatively large differences in performances: For coupon 2, in the ensemble submitted by Team 2, only an 8 % share of the overall prediction coming from the *parRF* model was replaced by a larger share of the already prominent *econ* forecast (from 41 % to 50 %). This lead to a 9 % improvement (reduction in assessment criterion score) – an improvement that accounted for more than half of the difference between the two teams and thus the difference between the 6th and the 1st overall rank. Noteworthy is the fact that in the test set comparisons, Team 1's ensemble had performed better, if only by the very small margin of 0.2 %.

The fact that our seemingly second-best submission by Team 2 outperformed our "best guess" submitted by Team 1 was surprising, while not inexplicable. Every ensemble forecast submitted by Team 1 performed better on the test set than the respecting forecasts by Team 2. However, the ensembles in the eventual winning contribution performed only slightly worse on the test set and much better on the classification set in two out of three cases. This leads to the assumption that some of the ensembles found by the random subspace method were in fact overfitting and performed better on the "lucky" test set sample than on the "unlucky" classification data:

| Submission | Overall Rank | T1 | T2 | T3 | BV | Total |
|---|---|---|---|---|---|---|
| Team 2 | **1** | 2647.6 | 3159.7 | 4196.9 | 1167.2 | **11171.3** |
| Team 1 | **6** | 2637.9 | 3477.0 | 4323.9 | 1167.2 | **11605.9** |
| Difference | **5** | -9.8 | 317.3 | 127.0 | 0.0 | **434.6** |

**Table 5.2**: Overall performance of submissions (by loss function values)

As can be identified in table 5.1, the ensembles submitted by Team 2 were the 2nd, 1st, and 1st best performing ensembles for the three coupon redemption probabilities on the classification set (and the 1st best for the basket value; not shown in the table).

When considering the ensembles for our second submission, we tried to choose combinations that performed slightly worse, but were safer bets – and we were very fortunate with that approach.

# 6 CONCLUSION

In this paper, our goal was to examine the main factors responsible for the team of the Humboldt-University Berlin winning the Data Mining Cup competition 2015. Even without the exact solutions of the other teams available, we were able to identify factors that differentiated our second, winning submission, from the first, which came sixth. Predictions for the basket value relied on a model mainly based on the median value of previous orders. This helped to predict the value of an extreme outlier more accurately. However, both submissions contained the same BV predictions. Also, the loss function yielded the same results for Coupon 1 for both submissions, leaving the main difference in the final overall score to the estimated redemption probabilities of coupon 2 and 3. The main difference here was the differing composition of the ensemble forecasts: submission two relied to a little extent more on a fine-tuned random forest model, which performed very well also as an individual classifier.

Besides the identification of key success factors, the post-competition analysis offered a number of other highly valuable insights to us. Firstly, when it comes to the training of classifiers on the available data, fine tuning of meta-parameters can impact the overall result a lot. In our case, tree based classifiers were usually inferior to neural networks, but a single well-calibrated random forest model ultimately showed the best results. Secondly, the performance of each individual model on the test data set should not be overestimated. Classifiers ranking low at this stage are still able to perform reasonable on the data that is to be predicted, and vice versa. Hence, using an appropriate ensembling strategy is also vital to success, since the pooling of several estimations reduces the risk of relying on one that turns out to be a poor estimator for the classification set. However, more complex ensembling algorithms need not to result in better predictions, as our case shows, since a simple averaging strategy was superior to the more complex ensemble selection approach. Finally, if sufficient information about the data is available or can be obtained via other sources, it is worth developing alternative approaches in addition to the standard machine learning repertoire based on domain knowledge and human insight. Combined with other forecasting methods, even simple models can influence forecasts favorably.

# 7 APPENDIX

**Table 4.1:** Loss function values of classification models trained on data set **version 6.3** per coupon

| Class Data Rank | Base Model | AC value Coupon 1 | Test Data Rank | Base Model | AC value Coupon 2 | Test Data Rank | Base Model | AC value Coupon 3 | Test Data Rank |
|---|---|---|---|---|---|---|---|---|---|
| **1** | econ | 2687.77715 | 11 | econ | 3140.2752 | 2 | nnet | 4338.76428 | 2 |
| **2** | nnet | 2747.2391 | 1 | avNNet | 3476.97427 | 1 | avNNet | 4363.19153 | 1 |
| **3** | avNNet | 2811.52517 | 2 | mlp | 3599.19806 | 3 | econ | 4415.54073 | 3 |
| **4** | mlp | 2856.41863 | 17 | mlpWeightDecay | 3613.37595 | 4 | fda | 4550.0429 | 10 |
| **5** | svmRadialCost | 2859.07284 | 7 | gcvEarth | 3617.5039 | 24 | gcvEarth | 4558.24574 | 5 |
| **6** | cforest | 2863.71121 | 5 | cforest | 3617.99332 | 5 | svmRadial | 4597.04679 | 22 |
| **7** | svmRadial | 2868.54697 | 6 | svmLinear | 3625.66027 | 13 | svmLinear | 4597.99714 | 15 |
| **8** | mlpWeightDecay | 2873.99956 | 20 | C5.0 | 3625.73504 | 12 | cforest | 4598.60065 | 12 |
| **9** | svmPoly | 2897.03921 | 12 | C5.0Tree | 3625.73504 | 11 | svmRadialCost | 4598.65374 | 21 |
| **10** | svmLinear | 2897.29704 | 13 | svmPoly | 3625.79811 | 8 | svmPoly | 4599.01563 | 18 |
| **11** | C5.0 | 2897.56605 | 15 | multinom | 3627.06262 | 19 | C5.0Tree | 4600.56209 | 16 |
| **12** | C5.0Tree | 2897.56605 | 14 | svmRadial | 3629.20378 | 15 | C5.0 | 4600.56209 | 17 |
| **13** | pcaNNet | 2909.85981 | 9 | nnet | 3629.2202 | 7 | pcaNNet | 4643.54309 | 4 |
| **14** | rf | 2911.1392 | 4 | pcaNNet | 3629.6606 | 25 | mlp | 4646.14745 | 14 |
| **15** | parRF | 2911.18204 | 3 | svmRadialCost | 3630.52264 | 20 | multinom | 4661.87802 | 6 |
| **16** | multinom | 2919.65654 | 23 | fda | 3636.77872 | 16 | mlpWeightDecay | 4668.28933 | 20 |
| **17** | rbfDDA | 3090.04362 | 22 | rf | 4052.24888 | 17 | knn | 4870.08609 | 28 |
| **18** | RRFglobal | 3123.78875 | 10 | parRF | 4104.54933 | 6 | rf | 4890.27127 | 19 |
| **19** | RRF | 3158.47613 | 21 | rbfDDA | 4262.48571 | 29 | parRF | 4930.18093 | 13 |
| **20** | treebag | 3167.18572 | 28 | treebag | 4416.94196 | 28 | rbfDDA | 5204.19767 | 31 |
| **21** | fda | 3182.00862 | 19 | RRF | 4637.03594 | 27 | RRFglobal | 5434.23072 | 26 |
| **22** | knn | 3196.94014 | 27 | RRFglobal | 4690.02927 | 26 | treebag | 5488.85518 | 27 |
| **23** | gcvEarth | 3275.93965 | 8 | knn | - | 30 | RRF | 5496.90402 | 25 |

**Table 4.2:** Loss function values of classification models trained on data set **version 3.2** per coupon

| Class Data Rank | **Base Model** | AC value Coupon 1 | **Base Model** | AC value Coupon 2 | **Base Model** | AC value Coupon 3 |
|---|---|---|---|---|---|---|
| 1 | mlp | 2856.377202 | avNNet | 3669.092155 | mlp | 3593.697531 |
| 2 | rbfDDA | 2862.538665 | mlp | 3685.585254 | mlpWeightDecay | 3594.239214 |
| 3 | mlpWeightDecay | 2872.270869 | mlpWeightDecay | 3697.646628 | svmRadial | 3653.251827 |
| 4 | knn | 2955.600623 | nnet | 3749.428752 | avNNet | 3669.266458 |
| 5 | avNNet | 2987.300624 | rbfDDA | 4163.78194 | nnet | 3674.695291 |
| 6 | nnet | 2988.694791 | pcaNNet | 4769.341142 | svmRadialCost | 3700.576407 |
| 7 | parRF | 4258.429395 | parRF | 4966.501562 | knn | 3861.376036 |
| 8 | rf | 4302.17466 | C5.0 | 4997.444635 | svmLinear | 4188.014667 |
| 9 | pcaNNet | 4368.552275 | rf | 5006.279729 | rbfDDA | 4260.756127 |
| 10 | cforest | 4420.633243 | gcvEarth | 5052.803812 | pcaNNet | 4761.347854 |
| 11 | gcvEarth | 4420.893701 | multinom | 5135.046115 | C5.0Tree | 4825.785503 |
| 12 | C5.0Tree | 4439.549251 | cforest | 5160.9559 | gcvEarth | 4871.221732 |
| 13 | C5.0 | 4459.080451 | C5.0Tree | 5209.192086 | C5.0 | 4880.7593 |
| 14 | RRFglobal | 4469.627573 | RRFglobal | 5257.812339 | cforest | 4883.387894 |
| 15 | RRF | 4514.835757 | RRF | 5278.575597 | parRF | 4947.413421 |
| 16 | multinom | 4540.900391 | treebag | 5407.787209 | rf | 4954.042926 |
| 17 | treebag | 4602.35559 | knn | 5527.243659 | multinom | 4970.120857 |
| 18 | fda | 4720.185743 | fda | 5625.194766 | svmPoly | 5044.922788 |
| 19 | econ | - | econ | - | fda | 5280.949221 |
| 20 | svmRadialCost | - | svmRadialCost | - | RRF | 5369.216666 |
| 21 | svmRadial | - | svmRadial | - | treebag | 5385.703474 |
| 22 | svmPoly | - | svmPoly | - | RRFglobal | 5417.34768 |
| 23 | svmLinear | - | svmLinear | - | econ | - |

**Table 4.3:** Loss function values of classification models trained on data set **version 1.0** per coupon

| Class Data Rank | **Base Model** | AC value Coupon 1 | **Base Model** | AC value Coupon 2 | **Base Model** | AC value Coupon 3 |
|---|---|---|---|---|---|---|
| **1** | mlp | 2856.418628 | mlp | 3599.198059 | fda | 4550.042903 |
| **2** | svmRadialCost | 2859.072839 | mlpWeightDecay | 3613.375949 | gcvEarth | 4558.245739 |
| **3** | cforest | 2863.711205 | gcvEarth | 3617.503902 | svmRadial | 4597.046789 |
| **4** | svmRadial | 2868.546966 | cforest | 3617.993316 | svmLinear | 4597.997142 |
| **5** | mlpWeightDecay | 2873.999555 | avNNet | 3620.419764 | cforest | 4598.600645 |
| **6** | avNNet | 2889.093697 | svmLinear | 3625.660266 | svmRadialCost | 4598.653737 |
| **7** | svmPoly | 2897.039209 | nnet | 3625.734932 | svmPoly | 4599.01563 |
| **8** | svmLinear | 2897.297038 | C5.0 | 3625.735043 | C5.0Tree | 4600.562089 |
| **9** | C5.0 | 2897.566047 | C5.0Tree | 3625.735043 | C5.0 | 4600.562089 |
| **10** | C5.0Tree | 2897.566047 | svmPoly | 3625.798108 | nnet | 4600.781484 |
| **11** | nnet | 2899.735544 | multinom | 3627.062622 | avNNet | 4607.922326 |
| **12** | pcaNNet | 2909.859813 | svmRadial | 3629.203782 | pcaNNet | 4643.543092 |
| **13** | rf | 2911.1392 | pcaNNet | 3629.660603 | mlp | 4658.851244 |
| **14** | parRF | 2911.182044 | svmRadialCost | 3630.522637 | multinom | 4661.878016 |
| **15** | multinom | 2919.656544 | fda | 3636.778716 | mlpWeightDecay | 4698.641147 |
| **16** | RRFglobal | 3123.788753 | knn | 3651.960438 | rf | 4890.271273 |
| **17** | RRF | 3158.476132 | rf | 4052.248877 | parRF | 4930.180933 |
| **18** | treebag | 3167.185715 | parRF | 4104.549333 | knn | 5191.497221 |
| **19** | fda | 3182.008619 | rbfDDA | 4262.485714 | rbfDDA | 5204.197674 |
| **20** | gcvEarth | 3275.939645 | treebag | 4416.941959 | RRFglobal | 5434.230715 |
| **21** | rbfDDA | 3363.178416 | RRF | 4637.035936 | treebag | 5488.855184 |
| **22** | knn | 3524.102362 | RRFglobal | 4690.029269 | RRF | 5496.904021 |

**Table 4.4:** Loss function values of base models trained on different data sets for basket value estimation

| Class Data Rank | **Base Model** | Data Set 5.2 | Data Set 6.3 | Test Data Rank |
|---|---|---|---|---|
| **1** | Median-based I | - | 1167.1772 | 1 |
| **2** | glmboost | - | 1261.17421 | 17 |
| **3** | svmPoly | 1602.04522 | 1285.40786 | 18 |
| **4** | Median-based II | - | 1467.05972 | 2 |
| **5** | cubist | - | 1652.7397 | 3 |
| **6** | qrf | - | 1737.97647 | 8 |
| **7** | RRFglobal | 1843.64988 | 1742.81696 | 7 |
| **8** | cforest | - | 1743.27874 | 9 |
| **9** | rf | 1844.1442 | 1743.41284 | 4 |
| **10** | RRF | 1842.9287 | 1743.65493 | 6 |
| **11** | parRF | 1841.89611 | 1744.29778 | 5 |
| **12** | ctree | 1835.59469 | 1749.99575 | 10 |
| **13** | blackboost | - | 1757.68262 | 11 |
| **14** | gbm | 1878.07381 | 1764.03542 | 12 |
| **15** | mlpWeightDecay | 1956.46216 | 1961.0433 | 13 |
| **16** | mlp | 1977.49238 | 1971.62002 | 14 |
| **17** | svmRadial | 1996.20736 | 1989.92665 | 15 |
| **18** | svmRadialCost | 1996.49866 | 1990.28565 | 16 |

**Table 5.1:** Loss function values of model combinations per coupon (submitted ensembles bold)

| Class Data Rank | Ensemble | AC value coupon 1 | Test Data Rank | Ensemble | AC value coupon 2 | Test Data Rank | Ensemble | AC value coupon 3 | Test Data Rank |
|---|---|---|---|---|---|---|---|---|---|
| 1 | **RE #18** | **2637.843942** | **1** | **Top 2** | **3159.712347** | **7** | **Top 3** | **4196.853209** | **5** |
| 2 | **RE #17** | **2647.594351** | **3** | RE #5 | 3219.600623 | 3 | RE #3 | 4321.524662 | 2 |
| 3 | RE #14 | 2664.224799 | 8 | RE #1 | 3330.800281 | 11 | **RE #4** | **4323.895534** | **1** |
| 4 | RE #12 | 2689.226412 | 10 | RE #3 | 3333.037064 | 8 | RE #5 | 4341.073662 | 4 |
| 5 | RE #10 | 2706.143125 | 13 | **RE #9** | **3476.974269** | **1** | Top 7 | 4350.467616 | 9 |
| 6 | RE #15 | 2717.759149 | 7 | RE #8 | 3476.974269 | 2 | RE #2 | 4392.201138 | 3 |
| 7 | RE #16 | 2717.759149 | 6 | Top 3 | 3525.389222 | 9 | RE #1 | 4571.293999 | 7 |
| 8 | Top 2 | 2739.544371 | 22 | Top 4 | 3543.771349 | 10 | | | |
| 9 | RE #13 | 2740.993811 | 9 | RE #2 | 3569.163531 | 13 | | | |
| 10 | RE #21 | 2747.239104 | 2 | | | | | | |
| 11 | Top 10 | 2816.475518 | 24 | | | | | | |
| 12 | Top 3 | 2905.847441 | 20 | | | | | | |
| 13 | Top 4 | 2905.847441 | 25 | | | | | | |
| 14 | Top 5 | 2905.847441 | 11 | | | | | | |