Data Mining Cup 2010 Report

Levatić, Jurica
University of Zagreb
Faculty of Science
Department of Mathematics

Malenica, Antonija
University of Zagreb
Faculty of Science
Department of Mathematics

Pavlic, Ilija
University of Zagreb
Faculty of Science
Department of Mathematics

June 17, 2010

Abstract

Data Mining Cup is the world's largest student data mining competition, organized by prudsys AG. It is held annualy. The task of 2010 was maximising revenue by intelligent couponing – some customers will decide to purchase again if given a discount coupon, resulting in a profit. On the other hand, we must not give the discount to those who will decide to purchase again regardless, as not to incur a loss.

In our report we describe our approach to the issues we faced in trying to classify customers as repurchasers and non-repurchasers using the data set of 32.429 instances (customer orders) described with 38 attributes.

To create the final classification model, we use a heuristic approach to try to improve the results of a Parallel Random Forest algorithm by fine-tuning and human intuition.

1 Introduction

Our task consisted of maximising revenue by intelligent couponing.

> *"Many customers only make an order in an online shop once. There are many reasons why they do not make another order. Online dealers try to counteract this using appropriate customer loyalty measures. For example, a tried-and-tested method is to hand out vouchers some time after an order to encourage the customer to make a follow-on purchase. But from an economic perspective sending vouchers to all customers is not a good solution because customers may make a second purchase without the incentive.*
>
> *Using the existing characteristics of a customer's initial order, such as order quantity per type of goods, title and delivery weight, a decision must be made on whether to send a voucher worth €5.00.*
>
> *The customers who receive a voucher should be those who would not have decided to re-order by themselves. The reason for that is based on empirical results — it should be assumed that the voucher initiates a purchase with an average order value of €20.00 in 10% of those not making a purchase.*
>
> *If a voucher is sent to a customer that would have re-purchased anyway this results in a €5.00 loss for the dealer.*
>
> *The aim is therefore to maximise revenue by sending the vouchers to selected customers [1]."*

This was formalized as a table giving outcomes depending on our decisions and actual results.

		Real	
		Non-repurchasers (0)	Repurchasers (1)
Forecast	No voucher (0)	0	0
	Voucher (1)	1.5	-5

Table 1: Cost matrix of outcomes from [1]

Given the following variables

$$\begin{aligned} i &= \text{Customer number} \\ g_i &= \text{Coupon decision} \\ k_i &= \text{Purchase} \end{aligned}$$

the **profit** was calculated using a formula

$$x_i = \begin{cases} 0 & g_i = 0 \\ -5 & g_i = 1 \wedge k_i = 1 \\ 1.5 & g_i = 1 \wedge k_i = 0 \end{cases}$$

The total profit is then

$$\sum_{i \in \text{customer numbers}} x_i$$

The task was presented as a data set of 32.429 previous purchases described with 38 attributes using which we had to decide whether to give a coupon or not.

1.1 Dataset

The dataset originally consisted of 38 attributes. We list the original attributes:

customernumber Unique customer number

date Date of first order

salutation Whether the customer is male, female or a company

title Whether the title is available

domain E-mail provider domain

datecreated Date the account was opened

newsletter Whether the customer is subscribed to a newsletter

paymenttype

deliverytype

invoicepostcode Invoice address postcode

voucher

advertisingdatacode

case Value of goods (1-5)

numberitems

gift Gift option

entry Entry into the shop

points Points redeemed

shippingcosts Whether the shipping cost was incurred

deliverydatepromised

deliverydatereal

weight Shipment weight

remi Number of remitted items

cancel Number of cancelled items

used Number of used items

w0-w10 Ten attributes enumerating the number of various types of items ordered

target90 Whether the customer re-ordered within 90 days

The data set contained a number of attributes which had most of the values missing. Also, there were some obvious outliers.

2 Our Approach

This section concerns with our approach to the problem, from getting to know the data set through the initial tests up to the final version of our model.

2.1 Initial tests

In order to better understand the given data set we conducted a series of experiments using various algorithms for classification as well as those for attribute selection.

Our goal was to find those attributes which give more information on customer reordering within 90 days.

Our approach was two-fold. Firstly, we performed tests using classification algorithms to see if some fared better than the others in correctly classifying our test set. As some algorithms usually do better with sets showing certain properties, apart from finding better algorithms *per se*, we could also find algorithms better-suited *to our dataset*, inferring some information of the data set attributes (for example, whether the data set is noisy).

Algorithms used in that manner include:

- Naive Bayes
- J48 decision tree
- K-Nearest Neighbours
- Boosted algorithm groups
- Bagging
- Random Forest
- Parallel Random Forest[1]

The most robust algorithm according to our tests was the **Parallel Random Forest** algorithm (further on referred as **PARF**).

Secondly, we used attribute selection algorithms in order to find attributes which carry more weight in our decision. Such algorithms were standard algorithms included with WEKA [5] were for example:

- ReliefF Attribute evaluation
- InfoGain Attribute evaluation
- Cfs Subset Evaluation

It is worth noting that PARF uses an attribute selection algorithm of its own. It is possible to use only k most-important attributes found in building the initial forest to rebuild the forest [4].

The test were usually performed using 10-fold cross validations or percentage splits, depending on the complexity of algorithms and the importance of the tests.

2.2 Modifying the data set

2.2.1 Balancing the classes

The initial set was unbalanced in regard to class – the ratio of "zeroes" and "ones" was approximately 4.3. The lack of positive examples proved to be a difficulty for most of the tested classifiers. Because of that we tried to somehow balance the number of instances belonging to positive and negative classes.

Some of the approaches we entertained were:

- Combining the set all of the instances of the positive class with a randomly sampled subset of negative classed instances of equal size
- Duplicating randomly chosen positive instances
- Creating new positive instances by combining existing positive instances [2, 3]

The test results were not promising, so we abandoned the idea of artificially balancing the classes. Instead of that we decided to use weighting on the classification in order to skew the classifications to our advantage.

[1] from Ruđer Bošković Institute, http://www.irb.hr/en/cir/projects/info/parf/

2.2.2 Creating new attributes

We acknowledged the difficulties of algorithms in discerning information present in the data set implicitly rather than explicitly.

Examples of such information are:

- The difference between promised and real delivery dates
- The amount of time it took the buyers to make their first order after registering
- Whether the items included the advertising codes (as opposed to what the codes actually are)
- The actual number of received items (#ordered - (#canceled + #remitted))
- Average item weight
- Whether the customers downloaded some items
- The proximity of public holidays

We created new attributes exposing such implicit information, and so increased the attribute count to 48.

In order to verify the quality of those new attributes, we used attribute selection algorithms. Newly introduced attributes ranked among the top 10 attributes in most of the attribute selection algorithms used.

2.3 Parallel Random Forest

The Random Forest is a well known algorithm. The Parallel Forest algorithm implementation from Institute of Ruđer Bošković[2] offers additional customization options like classification weighting and attribute ranking. Following the attribute ranking, the forest can be regrown using only the attributes with highest values. Additional details can be found in [4].

2.4 Heuristic approach

As previously noted, of all the tested algorithms, PARF proved to be the best choice for our dataset. We tried to think of a way to *help* or *guide* PARF using the experience gained during our work with the provided data.

To exploit our experience in obtaining better results, we have put forth the following two **working assumptions**:

1. There are some intuitive groupings of customers in respect to a small number of attributes.
2. Assuming a set divided into smaller groups of customers, it is possible to obtain better results by experimenting with different parameters of PARF algorithm and choosing the most appropriate ones

For example, it is likely for a customer which usually downloads and is also receiving a newsletter to repeat the order. Therefore, we could weight our decision heavily to avoid giving the coupon to re-purchasers.

As initial tests were promising, we have decided to push forward in using this approach.

3 Experiment and Discussion

In this section we describe in more detail the outcome of our approach.

With our working assumptions in mind, we developed an iterative approach to solving the given task. We noted that there is a **base profit** of around 9000 which can be attained automatically. The results of our tests had given us an **expected profit**.

1. Let S_0 be the set of all instances
2. If possible, find an intuitive attribute split of input set $S_{0,\ldots,k}$ to obtain sets

[2]Ruđer Josip Bošković (Dubrovnik, 18. May 1711. – Milano, 13. February 1787.), a theologian, physicist, astronomer, mathematician, philosopher, diplomat, poet, Jesuit.

$S_{i,\ldots,1}, \ldots, S_{i,\ldots,l}$, where k and l are numbers of attribute classes in their respective steps

3. For each of the newly generated sets, perform step 2 again

4. Test the algorithm parameters on all obtained subsets of S_0, and decide whether to use PARF (if so, also decide on the parameters) or classify all instances of the subset in a single class

Simply put, we have been building a decision tree in which the final subsets are leaves. We call the approach iterative because we use step 4 to verify our assumptions on the "intuitiveness" of our splits, and gain some form of feedback. We might try to backtrack if that would help the final solution. A section of the created tree is visible in Figure 1.

Additionally, we tested classification on the subsets using various parameters of PARF. We tried increasing classification weights and discerning the behaviour of profits. If we noticed that the profits rose up to a certain point then to start falling again we reasoned, that the weights at that point are probably close to optimum weights. The splitting facilitated testing of PARF parameters by decreasing of number of instances to be processed.

Finally, the expected profit is the sum of expected profits on all leaves.

3.1 Evaluation Criterion

The evaluation criterion of a model was a combination of a couple of factors:

- The intuitiveness of the attribute splits used to obtain the final sets
- The quality of classification results obtained by PARF
- If applicable, the difference in profits with different parameters of PARF

The final two points were measured quantitatively by expected profit, according to results achieved on the train set.

3.2 Experimental Results and Explanation

The following table contains tests performed on the extended data set of attributes, using the most promising algorithms according to tests performed on the original data set.

Algorithm	Expected profit	Percentage increase
k-NN	≈10000	11
PARF	≈11000	22
Heuristic PARF	≈11500	27

Table 2: Approximate values for expected profits according to our experiments

As can be seen from the table, our heuristic approach should give an 5% increase to the quality of classification.

The explanations we offer to that increase are:

assuming the expected increase is true

- Our intuition helped find good splits
- Smaller set sizes enabled us to grow larger forests, improving the classification results

assuming the expected increase is not true

- Overfitting
- Insufficient experiment repetitions coupled with cognitive bias produced overly optimistic expectations

4 Conclusion and Future Work

To summarize, we offer our conclusions in a list.

- Artificially balancing the classes *decreased* the quality of the solution in our tests. Most probable reasons for that are:
  1. Noisiness of the original set
  2. Insufficient knowledge on using the techniques appropriate to the given data set
- Producing new attributes helped improve the solution by ≈10%. Not all of the new attributes were of equal value for the increase in profit.
- Assuming our test exposed the differences correctly, using a heuristic approach coupled with PARF helped to further increase the profit for ≈5%.

In our further work we should use statistical analysis to check the assumptions laid out as parts of our conclusions. We could also try to work around the noisiness of the dataset.

5 Acknowledgements

We would like to thank Tomislav Šmuc and Matko Bošnjak for their helpful suggestions as well as assistance in compiling and using PARF.

References

[1] prudsys AG, Data-Mining-Cup
DMC 2010 Task,
http://www.data-mining-cup.de/en/dmc-competition/task/, 2010.

[2] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, W. Philip Kegemeyer
SMOTE: Synthetic Minority Oversampling Technique, Journal of Artificial Intelligence Research 16, 321–357, 2002.

[3] Sanjeev Suman, Kamlesh Laddhad, Unmesh Deshmukh
Methods for Handling Highly Skewed Datasets
http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.105.9574, 2005.

[4] Goran Topić, Tomislav Šmuc, Zorislav Šojat, Karolj Skala
Reimplementation of the Random Forest Algorithm, Parallel Numerics 119-125, 2005.

[5] The University of Waikato
WEKA Manual for Version 3-6-2,
http://www.cs.waikato.ac.nz/ml/weka/, 2010.

[6] Tomislav Šmuc, Matko Bošnjak, Dragan Gamberger,
Machine Learning lectures of Faculty of Science, Department of Mathematics at the University of Zagreb,
http://web.math.hr/nastava/su/materijali/, 2010.

[7] Various authors,
Wikipedia pages related to machine learning,
http://en.wikipedia.org/, 2010.

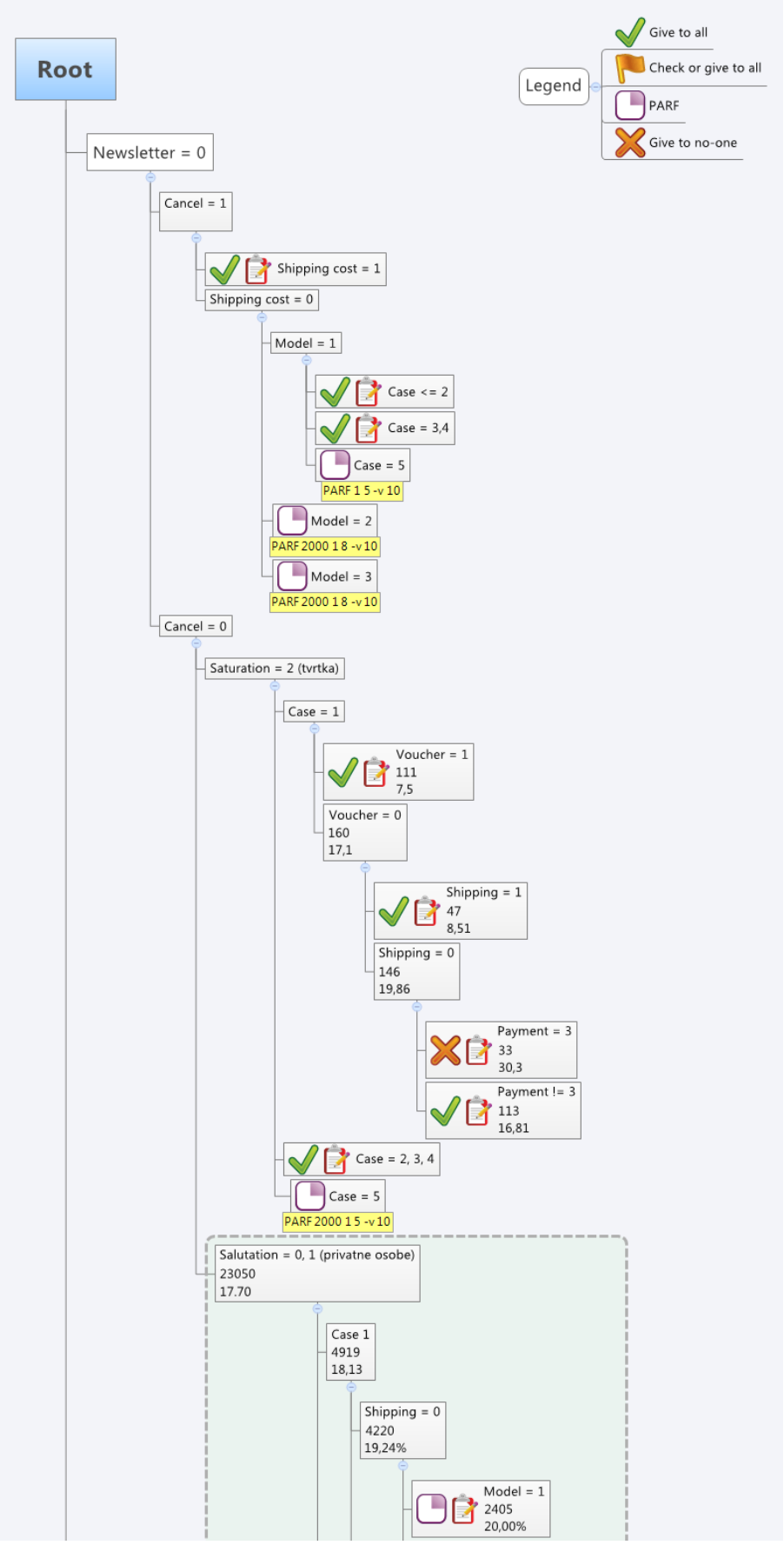

Figure 1: Section of the built decision tree