Estimation of minimum sample size for identification of the most important features: a case study providing a qualitative B2B sales data set

Marko Bohanec[1,*], Mirjana Kljajić Borštnar[2] and Marko Robnik-Šikonja[3]

[1] *Salvirt Ltd., Dunajska cesta 136, SL-1 000 Ljubljana, Slovenia*
E-mail: ⟨marko.bohanec@salvirt.com⟩

[2] *University of Maribor, Faculty of Organizational Sciences, Kidričeva cesta 55a, SL-4 000 Kranj, Slovenia*
E-mail: ⟨mirjana.kljajic@fov.uni-mb.si⟩

[3] *University of Ljubljana, Faculty of Computer and Information Science, Večna pot 113, SL-1 001 Ljubljana, Slovenia*
E-mail: ⟨marko.robnik@fri.uni-lj.si⟩

Abstract. An important task in machine learning is to reduce data set dimensionality, which in turn contributes to reducing computational load and data collection costs, while improving human understanding and interpretation of models. We introduce an operational guideline for determining the minimum number of instances sufficient to identify correct ranks of features with the highest impact. We conduct tests based on qualitative B2B sales forecasting data. The results show that a relatively small instance subset is sufficient for identifying the most important features when rank is not important.

Keywords: data set reduction, B2B sales forecasting, machine learning, sample size

Received: September 30, 2016; accepted: August 18, 2017; available online: November 30, 2017

DOI: 10.17535/crorr.2017.0033

1. Introduction

Practice has shown that application of machine learning (ML) models in business environments requires intensive education of business users even before the modeling. Many of the core ML concepts (like feature contribution to an outcome, data outliers, etc.) are new to audiences and require an explanation. Two initial steps require active participation from a business before ML modeling, specifically: a) nominating a list of initial descriptive features and b) collecting instances, as described by the nominated features. This inevitably raises two questions from business participants, which try to estimate the amount of additional work needed before machine learning

[*] Corresponding author.

models can be used: a) how many features are required? and b) the minimum number of instances to obtain useful information? Both questions have underlying expectations related to data acquisition costs and the users uncertainty concerning the utility of the effort. They represent the users implicit understanding that the general purpose of feature selection methods is to extract a small set of features that accurately classifies learning examples [3]. The stability of the selected features has been identified as an important aspect when the task is knowledge discovery, and not merely returning an accurate classifier [5]. Nogueira and Brown [6] has recently published a set of metrics to address feature stability in a more general way; however, given the setup of our experiment, this is not directly applicable.

While we analyzed the number of features in a business setting in our previous work [2], in this paper we estimate the minimum number of instances needed to learn important features. We use a publicly available B2B sales forecasting data set [1] as a case study.

The rest of the paper is organized as follows. In Section 2 we formalize an optimization problem. Section 3 discusses algorithm and results of experiments. Conclusions are put forward in Section 4.

2. Formalization of an optimization problem

Our goal is to find the smallest size of a random subset of instances V, which assures that for a given feature ranking function R, the ranks of the most important features remain the same for the entire data set.

$$R(a_1, ..., a_t) = R(s_1^V, ..., s_t^V),$$

R - ranking function,
a_i - rank of feature i on the complete data set of size n,
s_i^V - rank of feature i on a subset V of the data set.

We formalize an objective function, which we want to minimize.

$$f(|V|) = \arg \min_{|V|=1}^{n} \left[P((s_1^V, ..., s_t^V) = (a_1, ..., a_t)) \geq z\right], \tag{1}$$

z - detection probability for identical rankings on a random subset of size $|V|$ and on a complete data set of size n,
t - the number of top ranked features.

Note that in Eq. (1) we require the two ranks to be identical. However, for practical use, such a strict identity of ranks is not necessary. The mere identification of the top features is more significant than their ranks. For business discussions, the goal is to focus on the most important features and not to waste time and resources on less significant features. To reflect this requirement, Eq. (1) can be adjusted with an updated equality clause shown in Eq. (2), reflecting that any permutation of the top rank is sufficient.

$$f(|V|) = \arg \min_{|V|=1}^{n} [P(\exists perm(s_1^V, ..., s_t^V) = (a_1, ..., a_t)) \geq z]. \tag{2}$$

Function *perm* denotes any permutation of its arguments.

3. Experiments

In this section, we introduce the data set and ground truth for ranks, experimental algorithm, and analyze results.

3.1. Data set and ground truth ranks

We try to identify how the probability of identical ranks z defines the size of random samples $|V|$. We use a real world B2B sales data set with 448 instances, 22 features and a class feature with two values as described in [1]. To get ground truth rank of features $(a_1, ..., a_t)$, we rank features with a selected feature ranking algorithm on the complete data set. The ranks of features are presented in Table 1.

Rank	Feature	Impact
1	Up_sale	0,189
2	Client	0,184
3	Competitors	0,135
4	Source	0,107
5	Seller	0,077
6	Att_t_client	0,065
7	Posit_statm	0,044
8	Deal_type	0,034
9	Purch_dept	0,027
10	Budgt_alloc	0,025
11	Forml_tend	0,024
12	Comp_size	0,020
13	Product	0,017
14	Strat_deal	0,016
15	Scope	0,010
16	Partnership	0,006
17	Needs_def	0,003
18	RFI	0,000
19	Growth	-0,002
20	RFP	-0,004
21	Cross_sale	-0,005
22	Authority	-0,009

Table 1: *Rank of features using all instances and MDL evaluation function*

In this paper, we use the Minimum Description Length (MDL) evaluation function [4]. This measure favors features that compress the data well. We chose MDL because it is unbiased, fast and for our specific case gives similar rankings as the state-of-the-art algorithm ReliefF [7].

3.2. Experimental process

The goal is to measure how the upfront defined detection probability of equal ranks using random samples and whole data set (z from Eq. (1)) impacts the size of the samples. Intuitively, we expect that a larger z requires a larger $|V|$. To empirically evaluate this assumption, we repeated 30 iterations of each experiment for $z = 60\%$, 70%, 80% and 90% for scenarios reflecting the (non)importance of rank among the top features. An algorithm for experiments is described in Alg. 1.

Algorithm 1 *High level setup of the experiment*

```
procedure SubsetSizes(experiment parameters)
    for q in 1: numExperiments do
        $subsetSize = 0$
        repeat
            subsetSize = subsetSize + step
            correctRank = 0
            for  k in 1: sizeOfCompleteData do
                trialData = RandomSample(data, subsetSize)
                trialRank = RankEvaluation(trialData, $MDL$)
                correctRank = correctRank + Match(trialRank, groundTruthRank)
            end for
        until (correctRank $\geq$ requiredDetection)
        Store subset size
    end for
    Return stored subset sizes
end procedure                                ▷ Return table with results of all experiments
```

Mean and standard deviation of sample sizes obtained with experiments are reported in Table 2. Experiments are executed within the *R* environment.

	Top 1				Top 2			
	Rank		No Rank		Rank		No Rank	
Probability	Mean	SD	Mean	SD	Mean	SD	Mean	SD
60%	182,33	39,76	182,33	39,76	242,83	33,37	76,17	8,68
70%	350,50	35,14	350,50	35,14	357,33	35,62	101,17	10,14
80%	424,17	9,20	424,17	9,20	426,00	8,85	138,00	15,18
90%	437,67	7,85	437,67	7,85	440,17	3,59	177,00	19,01

	Top 3				Top 4			
	Rank		No Rank		Rank		No Rank	
Probability	Mean	SD	Mean	SD	Mean	SD	Mean	SD
60%	294,00	19,80	83,83	8,58	330,00	19,25	149,17	10,75
70%	372,83	28,79	111,67	10,93	387,17	14,78	184,33	9,35
80%	424,17	9,01	153,83	15,41	424,00	9,23	223,33	11,91
90%	441,17	2,84	199,00	25,13	441,33	3,70	263,17	23,58

Table 2: *Results of experiments for top 4 features: the number of instances required to detect top k features from Table 1*

3.3. Analysis of results

The columns of Table 2 are grouped as Top 1 to Top 4, where the number indicates the number of best features t from Eqs. (1) and (2). For example, the group Top 3 indicates that the Top 3 features from Table 1 are evaluated, specifically *Up_sale, Client and Competitors.*

Figure 1 shows a box and whisker plot of a set of experiments where the probability is set to 70% and the rank of top features is not important. To maintain the detection rate for the most important feature (*Up_sale*) the median sample size from the experiment is 355 instances (79% of complete data set). This shows that the most important feature is unstable and it takes almost all instances to recognize it as the top ranking feature on a random subset. On two occasions subsets were much smaller (see the two circles indicating outliers).

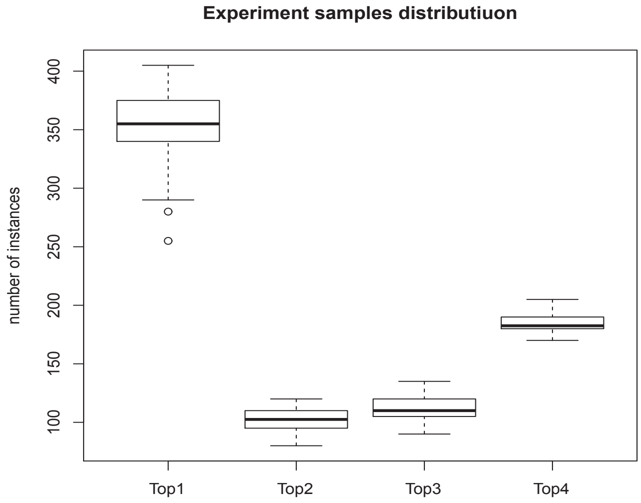
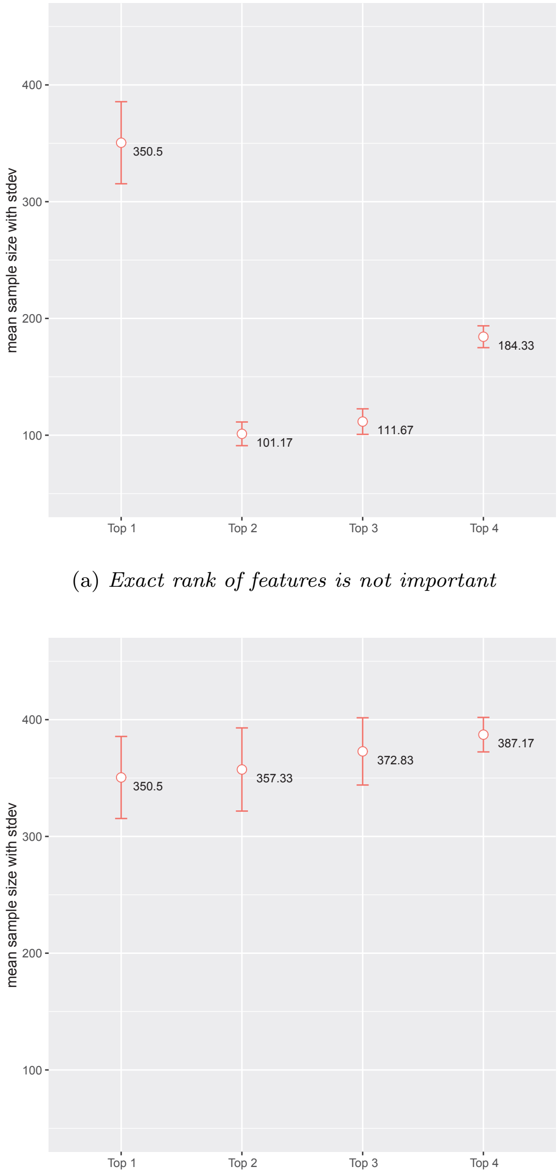


Figure 1: *Box and whisker graph for number of instances to detect top features with 70% probability (rank is not important)*

The results for the Top 2 features (*Up_sale, Client*) show much lower numbers. The median for Top 2 is 102 (23% of the complete data set), with an approximately symmetrical distribution. The reason for the big drop between Top 1 and Top 2 median values is that the rank of the top 2 features were not taken into account.

When comparing Top 2 and Top 3 (median 110), a slight increase in the number of samples is visible. This indicates that determination of Top 2 features is more stable and adding the third one adds some uncertainty - this can be explained by the scores of these features in Table 1. To analyze the impact of the rank, Table 2 compares the sample sizes when ranks are not taken into account ((Eq. 2)) and when they are considered (Eq.1). The most interesting values (i.e. small values of required instances) for business use are indicated with a gray background, in Table 2.

In Figure 2 two plots with a detection probability of 70% are shown: (a) ranks are not important and (b) ranks are important. In Figure 2a we see a drop of samples size after Top 1, the behavior which we observed in Figure 1. When looking at Figure 2b, which display statistics for experiments when the rank is important, we observe a different picture. For Top 1 Figures 2a and 2b show the same value, however in Figure 2b Top 2-4 exhibit higher mean subset size.



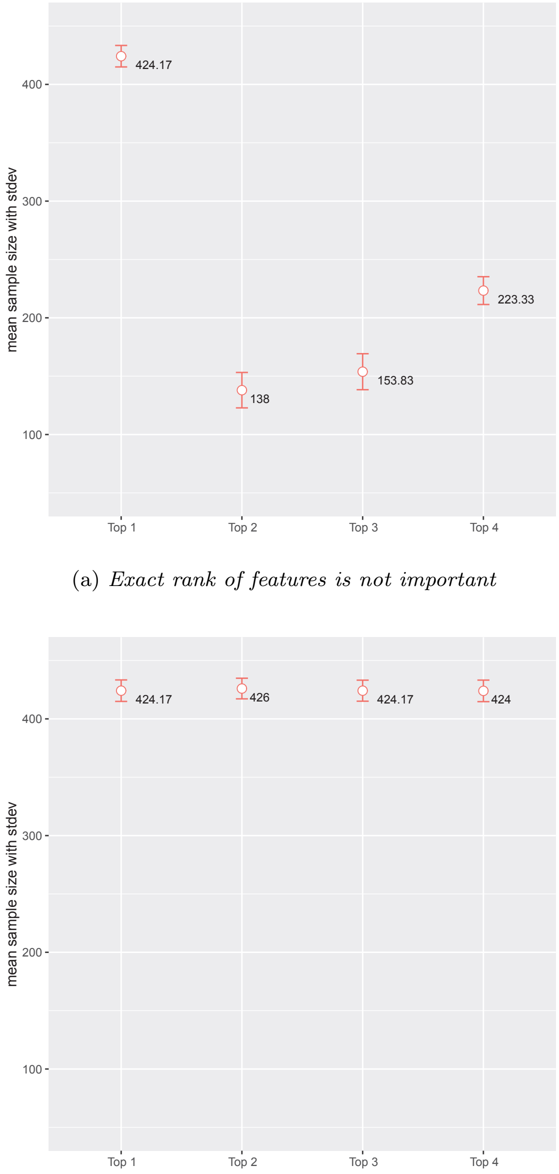
(a) *Exact rank of features is not important*

(b) *Exact rank of features is important*

Figure 2: *The number of instances which assures detection of the top feature with 70% probability.*

For example, in Figure 2b, to recognize Top 3 features with 70% probability and correct rank, on average 372 instances are required (83% of the complete data set). The value of standard deviation for Top 3 (28,79) indicates a slightly more stable performance for the three top features compared to the standard deviation for Top 1

(35,14). When comparing this with mean and standard deviation for Top 3 (10,93) from Figure 2a, we see the positive effect users might expect when flexible with the rank of features.



(a) *Exact rank of features is not important*

(b) *Exact rank of features is important*

Figure 3: *The number of instances which assures detection of the top feature with 80% probability*

Similarly, in Figure 3 we compare performance when (a) ranks are not important and (b) ranks are important and detection probability is set to 80%. Compared to 70%, all means are higher, while standard deviations remain comparable. When comparing Figures 2a and 3a, a jump in the mean value for Top 1 is visible, indicating a substantial increase in the average subset size (424, which is 94% of the total data set). For the rest of Top graphs in Figure 3a, a moderate increase in mean values is observed compared to Figure 2a. Depending on data acquisition costs, a 10% points increase in detection probability may be a reasonable tradeoff for a small increase in the number of instances.

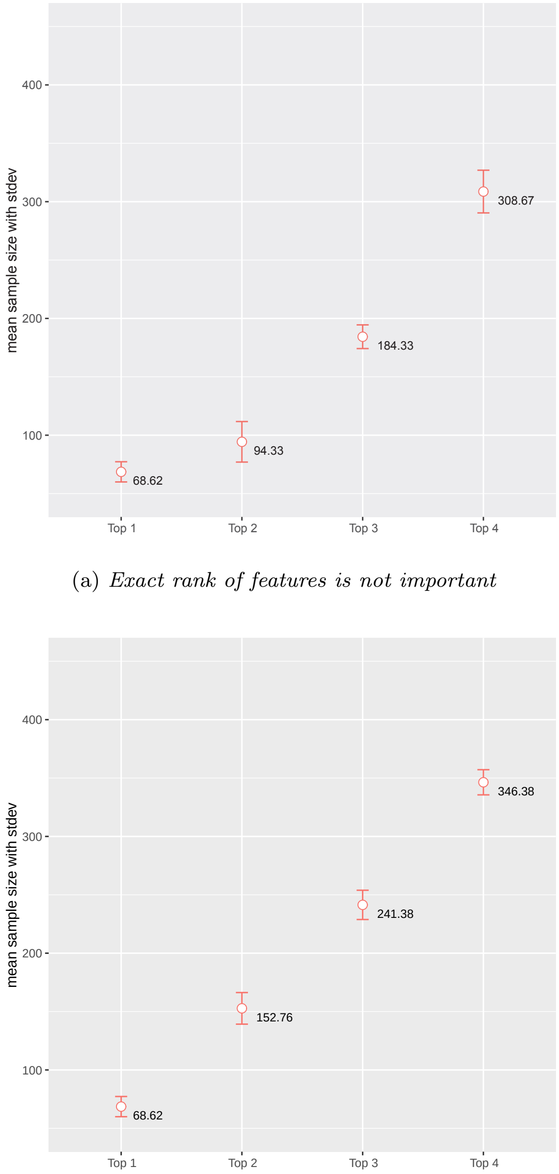
When exact ranks are required, the mean subset size is close to the size of the complete data set for all of the four most important features, as shown in Figure 3b. This is accompanied by a low standard deviation. When the rank of features is not important, the mean subset size for two or more top features exceeds 30% of the data, which makes a 80% detection probability less attractive for businesses.

If business users are satisfied with a detection probability of 60%, the results in Table 2 show that the mean for Top 3 features is 83,33 instances (18% of the complete data set), which encourages business users to engage in data collection task. In the same table, we show the results for a detection probability of 90%. When exact ranks are required, the mean subset size is close to the size of the entire data for all of the four most important features. This is accompanied by a low standard deviation. When the rank of features is not important, the mean subset size for two or more features exceeds 40% of the data, which makes this probability level less attractive for businesses that are considering to use ML techniques.

3.4. Analyzing the gap between groups Top 1 and Top 2

We further analyze a large drop between groups Top 1 and Top 2 when ranks are not important.

In Table 1 we notice that Top 2 features have a close impact value. We compute the linear correlation between these two features to be -0.74. This indicates a relatively strong negative correlation. To test this impact, we removed the second ranked feature *Client* from the data set and repeated 30 experiments using the detection accuracy $z = 70\%$ for both scenarios. The results are presented in Figure 4. When comparing Figures 2 and 4, we observe much smaller gaps between the feature groups. Note, that in Figure 4 features in all groups but Top 1 are different, as features gain one rank and feature *Seller* enters into the top 4 position.



(a) *Exact rank of features is not important*

(b) *Exact rank of features is important*

Figure 4: *The number of instances which assures detection of the top feature with 70% probability, without the feature Client*

4. Conclusions

For an interesting business case, we showcased how relatively small subsets of data can be used to detect the most important features with high probability. We also

showed that a significant reduction in the number of instances can be achieved when the ranks of top features are not important, otherwise, the gain is less noticeable. The findings are encouraging for business decision makers, as even without a large data set, they can focus on the most important features, even during the collection of data. Our findings can serve as a reference point and a guideline for implementation of ML techniques in qualitative B2B sales forecasting.
The focus of further research will be on using alternative ranking functions (i.e., ReliefF [7]) and other business data sets, where it is important to recognize the most important features as early as possible in order to improve the effectiveness of decision making. We will also analyze the correlation between features and their impact on the performance of decision models as well as their compactness and comprehensibility.

Acknowledgements

We are grateful to the company Salvirt Ltd. for funding the research and development of the optimization algorithm, used in this paper. Mirjana Kljaji Bortnar and Marko Robnik-ikonja were supported by the Slovenian Research Agency, ARRS, through research programs P5-0018 and P2-0209, respectively.

References

[1] M. Bohanec (2016). A public B2B data set used for qualitative sales forecasting research. Available at `http://www.salvirt.com/research/B2Bdataset/`.

[2] M. Bohanec, M. Kljajić Borštnar and M Robnik-Šikonja (2015). Feature subset selection for B2B sales forecasting. In Zadnik-Štirn, L. (Ed.). Proceedings of 13th International Symposium on Operational Research, SOR'15, Bled, Slovenia, 285–290.

[3] A. Kalousis, J. Prados and Melanie Hilario (2017). Stability of feature selection algorithms: a study on high-dimensional spaces. Knowledge and information systems 12(1), 95–116.

[4] I. Kononenko (1995). On biases in estimating multi-valued attributes. International Joint Conference on Artificial Intelligence, 1034–1040.

[5] L. I. Kuncheva (2007). A stability index for feature selection. Proceedings of the 25th IASTED International Multi-Conference Artificial Intelligence and Applications, Innsbruck, Austria, 421–427.

[6] S. Nogueira and G. Brown (2016). Measuring the Stability of Feature Selection. Springer International Publishing.

[7] M. Robnik-Šikonja and I. Kononenko (2003). Theoretical and empirical analysis of ReliefF and RReliefF. Machine learning 53(1-2), 23–69.