Transactions on Machine Learning
and Data Mining
Vol. 11, No 2 (2018) 63-77
© ISSN: 1865-6781 (Journal),
ISBN: 978-3-942952-65-1
ONLINE-ISSN: 2509-9337

ibai Publishing
www.ibai-publishing.org

Design of a Data Quality Simulation Application for Predictive Classification: Minimal Setup Viewpoint

Markus Vattulainen

Tampere University, Finland

markus.vattulainen@gmail.com

Abstract. Data quality simulations, controlled adding of data quality problems to data, are not common in industrial or scientific research reports on predictive classification. As a consequence, there is uncertainty regarding robustness of classification results achieved and what specific data quality dimension of the data production process should be improved. The best simulation applications have an extensive set of features but are limited by setup effort and expert level conceptual understanding required to run simulations. The current paper addresses a design question: what are the components of a data quality simulation application that requires no or minimal up-front setup effort? As a contribution, a component listing is presented and the feasibility of the design demonstrated by implementing it with R statistical language. Demonstration of the system with six business performance measurement system data sets suggests that controlled adding of eight common data quality problems (noise, missing values, low variance, outliers, class inconsistency, class imbalance, irrelevant features and low data volume) can be set up by a single line of R code enabling measurement of decrease in classification accuracy for each added data quality problem separately and in combination to support wider use of data quality simulations.

Keywords: Classification, Data quality, Simulation, System design

1 Introduction

Ensuring high and consistent quality of data [1] and gaining control of the cost elements involved [2] have been persistent problems in industry. As there can be data quality related variation in the data production process output, using data for its intended purpose such as predictive classification sets two questions: first, what happens to accuracy of results, including the possibility of failure in execution of the classification task, if data quality is not consistent over time and secondly, what is the

specific data quality dimension (e.g. completeness, accuracy, timeliness) data production process improvements should focus on given improvement cost constraints.

Data quality is a well-established field of research [3]. There is evidence that data quality dimensions can have main and interaction effects on predictive classification outcomes [4] and that effects of algorithms used for correcting data quality problems can be interdependent [5, 6]. Despite of frequent use of simulations in statistical research [7], data quality simulations are not common in industrial or scientific papers on predictive classification.

Data quality simulation for predictive classification is defined here as controlled adding of data quality problems to data to observe the impact of data quality problems separately and in combination on classification goodness measure (e.g. classification accuracy, kappa, sensitivity, specificity). In the terminology of design of experiment, classification goodness measure is the dependent variable and data quality problems independent variables. In the simulation the presence and intensity of each data quality problem can be fully controlled treatments as they are synthetically generated, intervening factors are blocked and since the simulation can use the same data that is used in the actual data mining task, high real word relevance of results is possible. The term 'simulation' is used instead of the term 'experiment', because the aim is to reproduce variation that is expected to happen in real-life data production processes and no specific hypothesis is being tested.

The obstacles to using simulations are attributed to use of in-house scripts instead of ready applications and uncertainty of correct simulation design [7]. The best current applications such as [8] provide an extensive set of features, but are limited by scripting and expert level understanding of statistical simulation concepts required. Thus, the design problem addressed in this paper is: what are the components of a data quality simulation application that requires no or minimal up-front setup effort? The design problem is of type: can we simplify an existing solution?

The objective of the current paper is to establish by demonstration an example design for a class of applications to be built in the category of easy to use data quality simulation applications. There are three limitations to the objective: first, the focus is exclusively on design of easily usable system and not on other solution criteria such as justifiability (i.e. how abstract data quality dimensions such as accuracy or completeness are operationalized as problems to be added to data for simulation and how the goodness of this kind of operationalizations can be evaluated) or feature completeness. Secondly, focus is set on design of components of the system and not on the production readiness of the system implementation. The implementation was built to demonstrate the feasibility of the design only. Thirdly, the scope includes controlled decrease of data quality of the original data set and possible extrapolation from that. In contrast, increase of data quality of the original set, which can be based on preprocessing combinations, is discussed in [9].

As the problem concerns design of an innovation [10], design research was selected as an approach. In the building phase, related research was surveyed from the specific point of view of constructing user requirements for the system. System components were identified. In the evaluation phase the proposed system was implemented

and simulation conducted with six business performance measurement system data sets.

As a contribution, the current paper presents components of a data quality simulation application and contributes to the existing literature on easy to use data quality simulation application design by presenting:

- a problem domain understanding including 10 user requirements
- a solution domain model as component listing
- an instantiation of the design as freely available R package "preprosim"
- demonstration of the system in the business performance measurement system domain.

The contribution is of incremental type as a specific limitation, easiness of setting up the simulations, in the current applications is addressed. The intended audience for the paper are data miners who do not yet use data quality simulations as part of their applied data mining research or industrial practice and who are interested in building data quality simulation to their data mining pipeline.

Section 2 reviews the related literature. Section 3 describes the method. Section 4 identifies the user requirements. Section 5 presents the solution model. Section 6 demonstrates the system with six cases from the business performance measurement system domain and section 7 discusses the implications, limitations and further research needs.

2 Related research

Related research was surveyed with the intention of finding the most fundamental concepts needed to build a data quality simulation application. As a limitation, statistical simulation literature without application building context was not included and no feature comparison of existing application was done. The earlier contributions were categorized to three categories and each category was used to construct user requirements to be presented in the design problem section.

Design of software such as data quality simulation software is a specific class of design problems. Chambers [11] defines the mission for statistical software as: "to enable the best and most thorough exploration of data possible. That means that users of the software must be able to ask the meaningful questions about their applications, quickly and flexibly.". For the purpose of designing a data quality simulation application, the contributions in the related research were classified to three categories: on the highest abstraction level there are conceptual foundations for data quality simulation, on the middle abstraction level there are empirical regularities found in descriptive data quality research studies and on the abstraction level closest to application design there are closely related statistical simulation applications as presented in Figure 1 (reading from top to bottom).

Conceptual foundations for data quality simulation consist of concepts of data quality and simulation. Data quality has been conceptualized as accuracy of represen-

tation of reality in the information system and as fitness for purpose [1]. The latter definition is adopted for data quality simulations and operationalized as classification goodness measure such accuracy or kappa. An important additional aim in conceptual data quality studies is to identify and group data quality dimensions such as representativeness and timeliness.

Simulations operate on generation of artificial data [12] or partially synthetic data [13]. Both produce effects that have two properties: first, effects are not in the original data and secondly, the effects are a known truth as they are generated in controlled manner in the simulation process. Data quality simulation adopts partially synthetic data approach and the system design task is to support generation of common data quality problems of various types and intensities. As an example, missing values, an operationalization of abstract data quality dimension 'completeness', can be added to data in various frequencies starting from no added missing values and ending to data with almost all values missing. Many data quality dimensions are non-trivial to specify as synthetic effects to be

added to data ("effect specification").

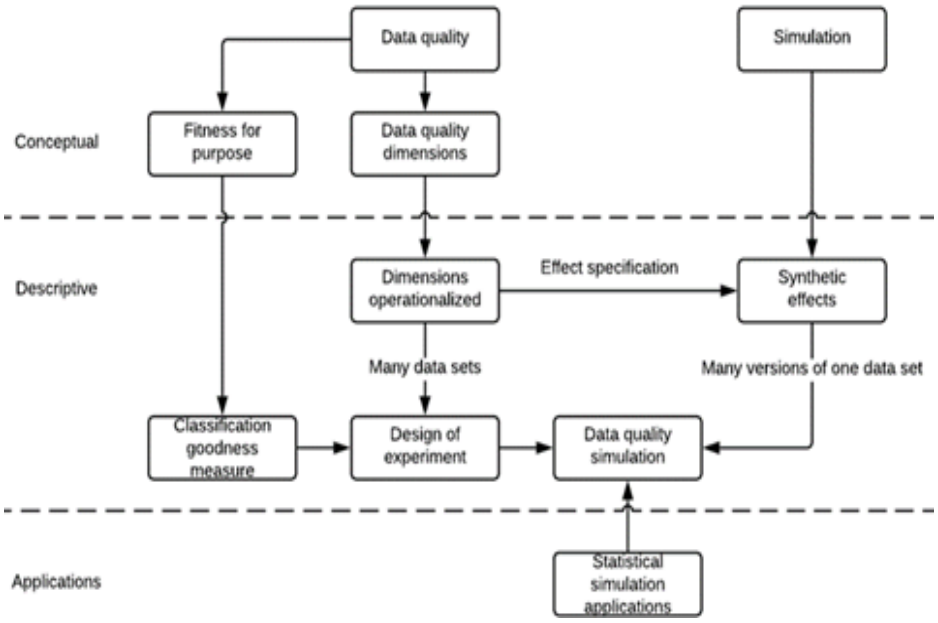

Fig. 1. Related research

Descriptive data quality research studies take data quality dimensions and ask first, how can a data quality dimension be operationalized as a measurable variable [14-16] and given the operationalization, what kind of empirical generalizations ("many data sets") can be found between the operationalization and data mining outcomes [4] (e.g. relationship between frequency of missing values and classification accuracy). Eight data quality dimensions and their operationalization from literature were selected as presented in Table 1. Technical specifications on how the selected

operationalizations were implemented as synthetic effects in R source code can be found in [17].

Table 1. Data quality problems to be added to data in simulation

Dimension	Operationalization	Studies	Effect specification
Completeness	Missing values	[12, 16]	Add missing values randomly
	Class imbalance	[18]	Add class imbalance
Accuracy	Noise	[19,20]	Add noise
	Low variance	[21]	Reduce variance
	Outliers	[22]	Add outliers
Consistency	Class inconsistency	[4, 15]	Swap class labels
	Irrelevant features	[23]	Add irrelevant features
Volume	Data volume	[19]	Undersample

In descriptive studies presence and significance of regularities can be quantified by design of experiment or visually observed by plotting. A corresponding data quality system design objective is to enable the finding of regularities between data quality problems and accuracy within a single data set ("many versions of one data set") by controlled manipulation (that is, creating versions of the original data set) instead of finding regularizes between several data sets. Also, instead of using a pre-selected classifier as in descriptive studies, data quality simulation application should support selection of classifiers via external libraries such as Caret [21] in the R statistical language.

Lastly, and closest to the task of designing a data quality simulation application are statistical simulation application descriptions such as [8]. Statistical simulation software has a broader and deeper scope than data quality simulation including generation of fully artificial data being drawn from known distributions. The best existing solutions have an extensive set of features but also three main limitations: first, scripting is required in the system configurations and specifically in the setup phase. Secondly, expert level understanding of statistical simulation concepts is required. Thirdly, the number of pre-configured data quality problems is small and can be limited to missing values and outliers only.

3 Method

Design research aims to contribute to the body of knowledge concerning the design of a class of applications and not only to produce useful applications [24]. Design research differentiates from routine design by method of construction and method of evaluation. The method of construction must be ideally grounded in systematic empirical findings, which are represented here by the related research, and be based on an explicit theory of either problem or solution domains [25]. Design research aims to validate design artifacts, represented here as component listing in Table 3., by presenting the design search space and by describing the design process so that it can be

reproduced. Rigorous evaluation should be conducted. [26] suggests a five-step design process: 1. Construct a conceptual framework, 2. Develop system architecture, 3. Analyze and design the system, 4. Build the [prototype] system and 5. Observe and evaluate the system.

The current study takes the eight data quality problems (Table 1.) as fundamental concepts in the problem domain (step 1) and 10 user requirements are identified. For the system architecture four components were designed (step 2). The detailed design was done by using object-oriented software design methods [27]. Specifically, prototype system [17] was designed with S4 object system [11] (step 3) and implemented with publicly available source code (step 4) for reproduction on the results. The system was observed by running it against six data sets from the business performance measurement system domain (step 5).

As an application domain, current business performance measurement systems aim to predict future instead of reporting what has happened in the past [28]. The aim is difficult to achieve as critics have pointed out [29]. Prediction is based on specification of measurements, gathering and collection of data and analysis of it [30]. Six real business performance measurement system data sets (Toyota Material Handling Finland, 3StepIT, M-Files, Innolink, Papua, Lempesti) are used to demonstrate the suggested system design. All of the cases have one financial target feature such as customer account profit margins and several non-financial predictors. Additional details of the data sets can be seen in Table 5.

4 Design problem

Design problem section focuses on understanding of the problem domain. The aim is to support application building effort and for this purpose problem domain description was done on the level of constructed user requirements as oppose to conceptual modelling of the problem. Construction of the user requirements (Table 2.) was done based on concepts found in related research and focuses mostly on design of experiment (DOE) as a foundational concept for data quality simulation.

Table 2. User requirements

Nro	User requirement	Concept
1	I can run data quality simulation on my data set and observe the change in classification accuracy when a specific data quality problem is added in varying intensities to all variables.	DOE, one independent variable, classification goodness measure
2	I can run data quality simulation on my data set and observe the change in classification accuracy when a specific data quality problem is added in varying intensities to a specific variable.	DOE, one independent variable, classification goodness measure
3	I can run data quality simulation on my data set and observe the change in classification accuracy when several data quality problem are added in varying intensities to all variables in a specific order	DOE, several independent variables, classification goodness measure

4	I can run data quality simulation on my data set without any system configurations and observe the change in classification accuracy when several data quality problems are added in varying intensities to all variables.	Simulation application, setup effort
5	I can run data quality simulation on my data set without any system configurations and observe observe the change in variable importance when several data quality problems are added in varying intensities to all variables	Simulation application, setup effort, variable importance
6	I can decide, which classifier to use and use several classifiers simultaneously.	Classification
7	I can use classifiers from an external classifier library.	Classification, setup effort
8	I can decide the number of holdout rounds for classification accuracy computation.	Classification goodness measure
9	I can run the simulation using multiple processors in parallel.	Parallelism
10	I can plot the decrease of classification accuracy by one added data quality problem in groups of a second added data quality problem intensities.	Data quality visualization

5 Design

Description of the design includes system architecture as input-output (Figure 2.) and system components (Table 3). Detailed design includes classes and data members (Table 3).

System inputs and outputs are presented in Figure 2. System takes as input the data set to be simulated and a classifier. Data quality problem parameters are not mandatory as indicated by the slashed line box. This is possible, because the system generates pre-set combinations of parameters. In the implementation eight data quality problems are included each with ten intensity categories (e.g. adding 0%, 10%, 20%...missing values). As output the system provides classification accuracy for each data quality problem parameter combination. As optional output the simulated data sets can be extracted. This is mostly a mechanism to validate that the data quality problems have been correctly added to data. Lastly, if parallel processors are available, they are supported.

The black box “Data quality simulation” in Figure 2. contains the four system components presented in Table 3. The responsibility of the SetUp component is to store the synthetic data quality effect specifications in the ParameterClass (e.g. in R language add noise from normal distribution: rnorm(length(x), x, noiseparameter)). The other data members of ParameterClass are variables in the original data synthetic effects are applied to, the parameters of the effects (e.g. standard deviation in noise adding example above) and the order in which the effects are applied. For easiness of use, the most important member function in the SetUp component is the automatic setting of parameters. The DataClass contains the class labels as categorical and nu-

merical other variables. The Contamination component gets the effect specifications, parameters and data objects and creates first a grid of parameter combinations and secondly, executes (that is, adds the specified data quality problems to the data) the data quality problem parameter combinations. The output is a list of contaminated DataClass objects.

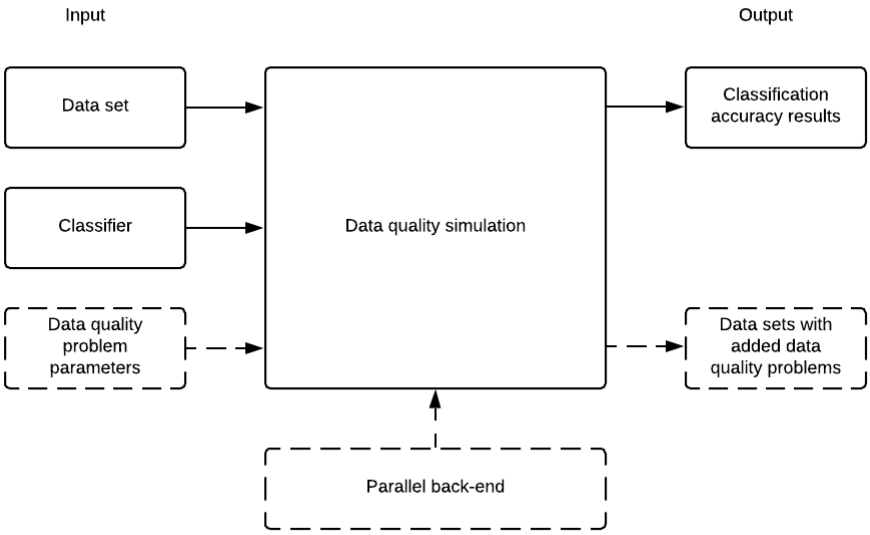

Fig. 2. Data quality simulation system inputs and outputs

Table 3. Components of a data quality simulation software with minimal setup viewpoint

Component	Class	Data member
SetUp	Parameter class	Data quality problem definitions
		Variables the original data set data quality problems are applied to
		Order of applying data quality problems
	Data class	Class labels
		Numerical variables
Contamination	Contamination	Grid of parameter combinations
		List of data class objects containing the simulated data sets
Evaluation	Evaluation	Vector of classification accuracies
Analysis		List of plots

Evaluation component gets the list and computes classification accuracy for each DataClass object. Table 4. presents an example intermediary output of this. The responsibility of the Analysis class is to extract data from the grid with classification accuracies and to output the various plots. Further analysis can be conducted to investigate for example the following: is there statistically significant difference in classification accuracy in the different categories of missing values when other data quality problem effects are set to constant and what kind of interaction effects, if any, are there between classification accuracy and missing values in different categories of added noise.

Table 4. An example of three data quality problems with three intensity categories and classification accuracy

Missing values	Noise	Swapped classes	Classification accuracy
0,0	0,1	0,1	0,94
0,1	0,0	0,1	0,75
0,1	0,1	0,0	0,83

Exploration of the design search space was outlined as a criterion for design research. The design is component-based as opposed to monolithic for two reasons: first, each of the components provides an interface supporting extraction of intermediary outputs for later use and possible alternative implementation of components, and secondly, the tasks of computing classification accuracies and analysis are by design to be supported by external libraries.

Efficiency decisions. The computations can be executed sequentially or in parallel depending on the availability of parallel backend. Both the creation of dataset versions and their classification goodness evaluations as in classification accuracy hold-out rounds are trivially parallelizable.

Memory use decisions. In the example implementation all the intermediate outputs and specifically the simulated data sets are stored in memory to be used by consequent components. This is memory-inefficient especially for big data and a production level implementation should store data set versions to external memory.

User interaction decisions. Parameter setting is the most important design issue regarding easiness of use. Rather that providing high-customizable simulation control object parameter combinations are automatically created from pre-set effect specifications and intensity levels.

Fault tolerance decisions. The user data entry points are data and parameter settings, both of which must include data validation mechanisms. Fault-tolerance weak point is the choice of classifier as many classifiers cannot handle either missing values, missing variance or other conditions introduced by adding of data quality problems.

6 Demonstration

Evaluation is one of the design research criterion as outlined in the method. The demonstration addresses three concerns: system output plots, execution time and easiness of use as R source code needed to setup the simulations.

6.1 Output plots

Output plots included three plots. First, to evaluate the robustness of classification accuracy as data quality varies boxplots of classification accuracies is presented with 6561 versions of each data sets. (Figure 3.). From the length of the range of values can be seen that adding of data quality problems has a substantial effect on classification accuracy in all the cases.

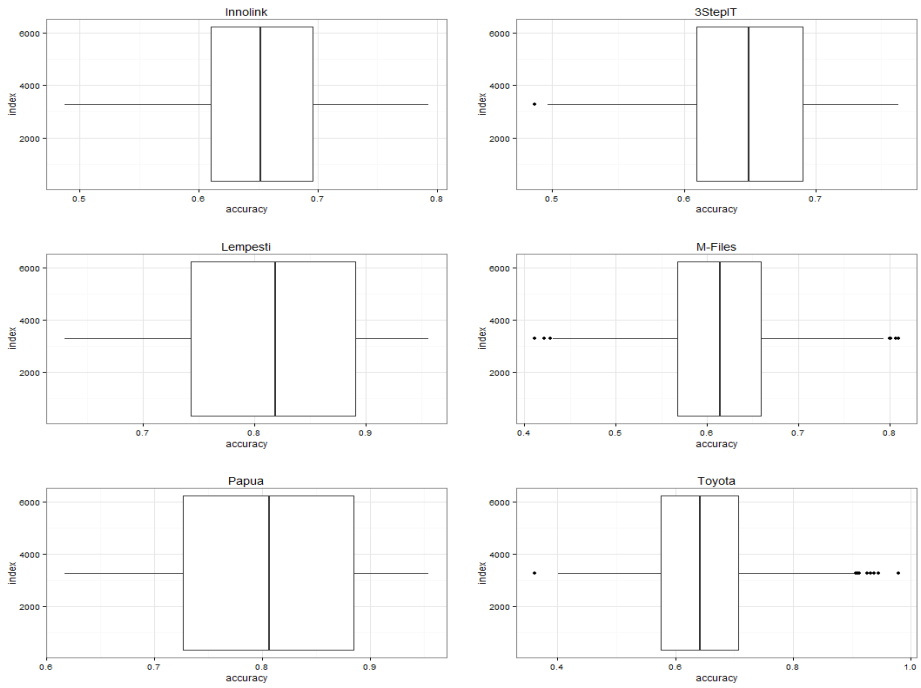

Fig. 3. Boxplot of classification accuracies, six cases

Secondly, to see how a specific data quality dimension affects classification accuracy Figure 4. shows the effect of increasing missing values in three groups of noise. From Figure 4. can be seen that the relationship between accuracy and missing values has similar kind of shape regardless of levels of noise introduced.

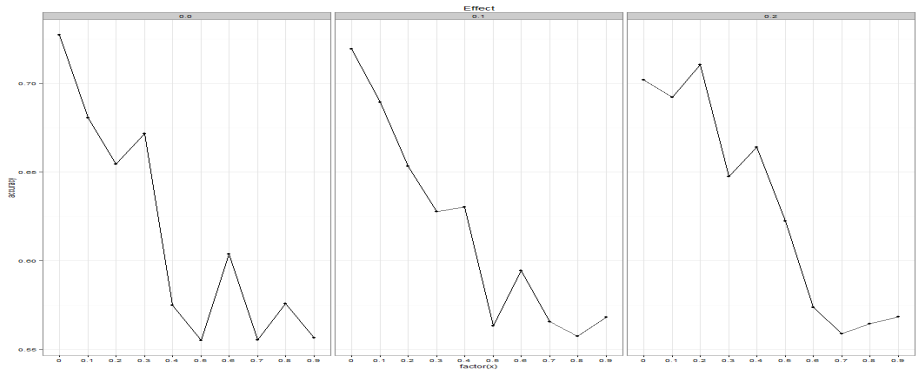

Fig. 4. Classification accuracy by missing values in three groups of noise

Thirdly, to estimate how robust the predictors are variable importance box plot is shown (Figure 5.). It can be seen that the most important variable is also the one having the largest amount of variation between contaminated data sets, which is as expected but can help to point out, which measurements should be focused on when improving the data production process.

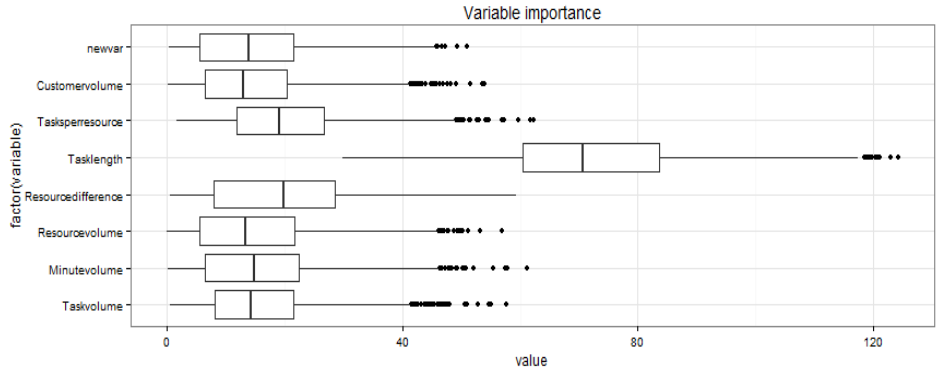

Fig. 5. Variable importance

6.2 Execution time

The execution time for each of the datasets (Table 5.) was computed using recursive partitioning tree from Rpart-package [31] or stochastic gradient boosting from Gbm-package [32] both of which were accessed via Caret [21]. Intel Celeron 1.6 GHz computer (2 logical processors) Intel i5-4200M 2,5 GHz (4 logical processors) and two different categories (16 and 32) holdout rounds with 70/30 percent training/test set divide were used for comparisons. All cases had 6561 versions of the original data set created.

Table 5. Executions times

Data set	n	p	Classi-fier	Process-sors	Holdout rounds	time (h)
Toyota MHF	48	23	Rpart	2	16	11
3StepIt	344	47	Gbm	2	32	23
M-Files	194	20	Gbm	4	16	4
Innolink	304	14	Gbm	2	32	23
Papua	253	10	Gbm	4	16	5
Lempesti	253	8	Gbm	4	16	4

6.3 Easiness of use

Boxplot of classification accuracy with 6561 versions of the input data can by done with a single line of R code. Parameter 'data' in the example is an R data frame with one class variable and other variables are numeric.

```
preprosimplot(preprosimrun(data))
```

In this function call 'preprosimplot' is part of the analysis component, 'preprosimrun' contains execution and evaluation components and 'data' is part of the setup component. Notice, that in the example above default data quality problem set and parameters are used.
Adding of a new classifier from caret classifier library.

```
preprosimplot(preprosimrun(data, caretmodel="gbm"))
```

The effect of adding missing values in three groups of noise:

```
res <- preprosimrun(data, type="xz", x="misval", z="noise")
preprosimplot(res, type="xz", x="misval", z="noise").
```

7 Discussion

The current study addressed a design question: what are the components of a data quality simulation application that requires no or minimal up-front setup effort? Eight common data quality problems were selected and design problem was understood by drawing from related literature resulting in 10 constructed user requirements for the system. The proposed system architecture was presented as a component table with four components: setup, contaminations, evaluation and analysis.

Six cases from business performance measurement system domain were used to demonstrate the design. Tentative evaluation is loosely based on March & Smith [33] evaluation criteria: effectiveness as system output plots, efficiency as execution time, generality and ease of use as number of lines of coding needed. As for effectiveness in a sense of whether the system is doing what it is intended to do, the system is able to compute variation in classification accuracy between data sets versions created in the simulation and plot what is the impact of one added data quality problem on classification accuracy in different groups of a second added data quality problem. Regarding efficiency, the system can make use of parallel processors and with the example data sets execution time was between 4 and 23 hours. For generality, it is argued that the data quality problems discussed (e.g. missing values, noise, outliers etc.) are not limited to business performance measurement system domain but are possible in other domains as well. Lastly, the system is easy to use measured by the amount of R code needed to setup a simulation run.

Research implications are that the proposed system model supports data set specific (i.e. many versions of one data set) exploration and quantification of the earlier generic descriptive findings (i.e. several data sets) on how data quality dimensions impact classification accuracy outcomes [4]. Compared to more extensive simulations application [8], the presented simplified approach aims to build foundations for higher adoption of data quality simulations in predictive classification research papers. Evidently, data quality simulation application should be not stand-alone but part of predictive classification pipeline starting from data acquisition, data preprocessing and following with classifier selection, classifier hyperparameter optimization, data quality simulation and classification goodness measurement.

Practical implications are that is it possible to build in an easy to use data quality simulation application and thus simulate data quality related variation in the data production process. The implementation can compute for an input data set how classification accuracy behaves when data quality is not consistent over time and identify how the data production process should be improved. That is, what is the specific data quality problem in the data production process that has the strongest impact on classification accuracy. For the purpose of implementing the design suggestions are:

- Define a set of data quality problems common for your application domain
- Support exploration of data quality problem parameter space by creating combinations of data quality problems with varying data quality problem intensity levels
- Build an interface to classifier library instead of hard-coded classifiers
- Build an interface to analysis and graphics libraries.

The implementation has several practical limitations. First, the implementation does not support flexible adding of new data quality problems to the system (i.e. "Contamination library"). As of author’s best knowledge this kind of library is not yet available. Secondly, the implementation does not provide numerical results such as ANOVA or any statistical significance testing capabilities. There is one significant research limitation: the paper does not evaluate or discuss the correctness of data quality problem specifications used as the focus is exclusively on how the setting up of simulations can be made easier.

Further research. Further research is needed to increase the number of data quality problems (to build the "Contamination library") that can be automatically applied in addition to the current eight data quality problems. This kind of library should also include validation of the synthetic effect specifications.

ACKNOWLEDGEMENTS. Professor emeritus Pertti Järvinen. After sales director Jarmo Laamanen Toyota Material Handling Finland, managing director Pekka Vuorela Innolink Group, sales director Mika Karjalainen 3StepIt, senior director Mika Javanainen M-Files, managing director Olli Vaaranen Papua Merchandising and managing director Sirpa Kauppila Lempesti for their support and the reviewers for their helpful comments.

References

1. Wand, Y., Wang, R.: Anchoring Data Quality Dimensions in Ontological Foundations, Communications of the ACM, Vol. 39, No. 11 (1996)
2. Eppler, M., Helfert, M.: A Classification and Analysis of Data Quality Costs, Proceedings of the Ninth International Conference on Information Quality (2004)
3. Sadig, S., Yeganeh, K., Induska, M.; 20 Years of Data Quality Research: Themes, Trends and Synergies, ADC '11 Proceedings of the Twenty-Second Australasian Database Conference - Volume 115, Pages 153-162 (2011)
4. Blake, R., Maggiameli, P.: The effects and interactions of data quality and problem complexity on data mining, Journal of Data and Information Quality, Volume 2, Issue 2, Article No. 8 (2016)
5. Engel, J., Gerretzen, J., Szymanka, E., Jansen, J., Downey, G., Blanchet, L., Buydens, L.: Breaking with trends in preprocessing, TrAC Trends in Analytical Chemistry, Volume 50, Pages 96–106 (2013)
6. Crone, S.F., Lessmann S., Stahlbock, R.: The impact of preprocessing on data mining: an evaluation of classifier sensitivity in direct marketing, European Journal of Operational Research Vol. 173:3, pp. 781–800 (2005)
7. Burton, A., Altman, D., Royston, P., Holder, R.: The design of simulation studies in medical statistics. Statistics in Medicine, 25(24), 4279–4292 (2006)
8. Alfons, A.: simFrame: Simulation Framework. R package, URL http://CRAN.R-project.org/package=simFrame. (2013) Accessed 17.6.2019
9. Vattulainen, M.: Preprocessing Optimization for Predictive Classification: Baseline Results from Six Industry Cases. Trans. MLDM 9(2): 48-61 (2016)
10. Järvinen, P.: On research methods, Opinpajan kirja, Finland (2012)
11. Chambers, J.: Software for Data Analysis, Springer, New York (2008)
12. Rubin, D.: Inference and missing data. Biometrika. 63 (3): 581–92 (1976)
13. Little, R.: Statistical Analysis of Masked Data, Journal of Official Statistics, 9, 407-426 (1993)
14. Pipino, L., Lee, Y., Wang, R.: Data quality assessment. Communications of the ACM, Volume 45 Issue 4, Pages 211-218 (2002)
15. Ballou, D., Pazer, H.: Modeling completeness versus consistency tradeoffs in information decision contexts, IEEE Transactions on Knowledge and Data Engineering, vol. 15, no. 1, pp. 240-243 (2003)

16. Shankaranarayanan, G., Cai, Y.: Supporting data quality management in decision-making, Decision Support Systems, vol. 42, no. 1, pp. 302-317 (2006)
17. Vattulainen, M.: preprosim: a lightweight data quality simulation, R package, URL http: //CRAN.R-project.org/package=preprosim. (2016) Accessed 17.6.2019.
18. Longadge, R., Dongre, S.S., Malik, L.: Class Imbalance Problem in Data Mining: Review, International Journal of Computer Science and Network,, Vol. 2, Issue 1 (2013)
19. Han, J., Kamber, M., Pei, J.: Data mining: Concepts and Techniques, Morgan Kaufmann, San Francisco (2012)
20. Xiong, H., Pandey, G., Steinbach, M., Kumar, V.: Enhancing data analysis with noise removal, *IEEE* Transactions on Knowledge and Data Engineering, vol. 18, no. 3, pp. 304-319 (2006)
21. Kuhn, M., Johnson, K.: Applied Predictive Modeling, Springer, New York (2013)
22. Kriegel H-P., Kröger P., Zimek, A.: Outlier detection techniqes, 16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC (2010)
23. Guyon, I., Elisseeff, A.: An Introduction to Variable and Feature Selection, Journal of Machine Learning Research. Vol 3, pp. 1157-1182 (2003)
24. Hevner, A., March, S., Park, J., Ram, S.: Design Science in Information Systems Resarch, MIS Quarterly, Vol. 28, No. 1, pp. 75-105 (2004)
25. Walls, J., Widmeyer, G., El Sawy, O.: Building an Information System Design Theory for Vigilant EIS, Information Systems Research (1994)
26. Nunamaker, J.F., Chen, M., Purdin, T.D.M.: Systems development in information systems research, Journal of Management Information Systems 7, No 3, 89-106 (1991)
27. Somerville, I.; Software Engineering, Pearson, Boston (2015)
28. Kaplan, R. S., Norton, D.P.: The Balanced Scorecard – Measures that Drive Performance, Harvard Business Review, Vol 71:1, pp. 71-9 (1992)
29. Nørreklit, H.: The balance on the balanced scorecard—a critical analysis of some of its assumptions, Management Accounting Research, Vol.11, Issue 1, pages 65–88 (2000)
30. Franco-Santos, M., Kennerley, M., Micheli, P., Martinez, V., Mason, S., Marr, B., Gray, D., Neely, A.: Towards a definition of a business performance measurement system, Internation Journal of Operation and Production Management, Vol 27:8, pp. 784-801 (2007)
31. Therneau, T., Atkinson, B., Ripley, B.: rpart: Recursive Partitioning and Regression Trees, R package, URL http: //CRAN.R-project.org/package=rpart. (2015) Accessed 17.6.2019.
32. Ridgeway, G.: gbm: Generalized Boosted Regression Models, R package, URL http: //CRAN.R-project.org/package=gbm. (2015) Accessed 17.6.2019.
33. March, S., Smith, G.: Design and natural science research on information technology, Journal Decision Support Systems, Vol. 15, Issue 4, pp. 251-266 (1995)