Using Parallel Sets for Visualizing Results of Machine Learning Based Plausibility Checks in Product Costing

ZANA VOSOUGH, SAP SE, Germany
VOLODYMYR VASYUTYNSKYY, SAP SE, Germany

Success in the business intelligence decision process is contingent on the ability to navigate, comprehend and validate large complex multidimensional data-sets. However, in many applications such as product costing there is often no tools that offer the kind of interactive visualizations necessary to make sense of the data. One particularly challenging problem is that of visualizing complex plausibility check reports produced by machine learning-based algorithms. In this work, we show how a real-world product costing validation tool developed as part of product costing application can be augmented with state-of-the-art visualization to facilitate visual analytics.

CCS Concepts: • **Human-centered computing** → **Information visualization**; • **Computing methodologies** → *Machine learning*; • **Applied computing** → Decision analysis;

Additional Key Words and Phrases: Parallel Sets, plausibility checks, product costing, big data

1 INTRODUCTION

Nowadays, the Big Data revolution is transforming how Business Intelligence applications acquire and process information [Chen et al. 2012]. A prerequisite for any subsequent analysis is the quality of the data. One obvious solution to this problem is manual maintenance and validation. However, this approach is unrealistic because it is too costly and too time consuming given the scale of the data. It is therefore of prime importance to develop automated means to quickly assess and validate big data. The recent development and stunning successes of machine learning offer sophisticated data processing algorithms, which can potentially help validate complex business data-sets at scale [Bose and Mahapatra 2001]. However, as we will show, these analysis results are often themselves large, complex, multidimensional and thus require novel means of analysis and interpretation. It follows that novel interactive visualization tools are needed to understand the output of machine learning algorithms applied to big data validation.

One enterprise application where data validity and reliability is extremely important is product costing. SAP Product Lifecycle Costing (PLC) application helps estimate the cost of a new product and which expenses will be incurred during the product's life-cycle. The large and detailed breakdown analysis of a product's cost – into potentially millions of items for complex products – helps make decisions on the product profitability and design by weighting costs against revenues. Furthermore, PLC applications can help a wide range of stakeholders (controllers, engineers, purchasers, etc.) in reducing the whole product's life cost. Importantly, PLC application users must assess not only the information presented to them, but also the confidence they have in that information. For example, there is often inaccurate or missing information in costing calculations because users make mistakes while entering data or lack information. PLC data validation – termed *plausibility checking* – is a critical issue since mistakes can have a dramatic impact on cost estimates and thus on business decisions. Plausibility checking is typically performed manually on projects containing hundreds of product cost estimates with up to millions of

Authors' addresses: Zana Vosough, SAP SE, Dresden, Germany, zana.vosough@sap.com; Volodymyr Vasyutynskyy, SAP SE, Dresden, Germany, volodymyr.vasyutynskyy@sap.com.

VisBIA 2018 – Workshop on Visual Interfaces for Big Data Environments in Industrial Applications. Co-located with AVI 2018 – International Conference on Advanced Visual Interfaces, Resort Riva del Sole, Castiglione della Pescaia, Grosseto (Italy), 29 May 2018

cost items - a tedious if not at times impossible task incurring high costs. Solutions to that are Machine Learning (ML) and data mining algorithms, which are in general a very effective approach to validate quality and automate plausibility checking for PLC data [Witten et al. 2016].

However, due to the scale of the data, plausibility checking will often return large number of potential errors, which makes their analysis and exploration challenging. What is needed is a visualization tool that would assist interactive exploration by providing (i) an overview over the different types of problems, (ii) a localization of problem areas, (iii) a dive-in view to explore details.
This paper is structured as follows:

- First, we introduce the industrial context of the project and the problem that we address.
- Second, we give details on the principle of operation and output of our machine learning-based plausibility check approach.
- Third, we describe the visualization method we used to represent the results of the machine learning.

2 PROBLEM STATEMENT

This section first describes the topic and research context and then explains the PLC's *plausibility check* problem.

2.1 Context

SAP Product Lifecycle Costing is a solution to calculate costs for new products or quotations. It helps to quickly identify cost drivers and to easily simulate and compare alternatives. PLC was developed in close collaboration with co-innovation customers who give regular input on the product ideas and prototypes over a period of four years [Vosough et al. 2016].

The initial trigger to develop a standard software for early product costing was a request from a customer. After this, we devised a development strategy that now spans 30 co-innovation customers that collaborate with us on specifications, requirements and evaluation [Vosough et al. 2017a]. The process of co-innovation allows for discussing new features and getting early feedback in a user-based design. Since approximately every quarter a new software version is released, and we run regular co-innovation workshops with customers in the same quarterly rhythm.

During the early phases of product costing, item prices are unknown or undefined for some time. As more data becomes available, the cost estimates are refined into new *versions* that eventually converge to a stable state. In this stable state, most parts of the product's cost structure can be delivered with a precise price fit to the desired costing goal. During this process, many decisions are made by experts such as controllers and engineers towards an estimation of the cost structure. This complex task and multiplication of individual contributors leads to data entry mistakes or wrong estimations. This makes the estimation of costs across the lifecycle of products extremely challenging to achieve. Errors are often uncovered too late and have a dramatic impact on the quality of cost estimates. In the following, we will describe the plausibility check process that can help detect common errors.

2.2 Plausibility Checks in Product Costing

Plausibility checks are intended to help users find potential errors by assessing the validity and plausibility of manual entries. The goal is to automate the detection of such errors to the largest extend possible. On the one hand, some trivial errors can be detected by simple hard-coded rules. For example, if a price for a material has not been set at all, a simple rule can detect cases where null values are present. On the other hand, there are plenty of less evident errors, the detection of which would require deep knowledge about the typical values and structures of the products. To detect such errors, we introduce the notion of *plausibility checks* that aim to detect potential errors such as anomalies, e.g. substantial deviations from typical values and structures.

Each type of plausibility checks is realized as a separate function producing its own validation messages. Further, the plausibility checks functions have the same input and output interfaces. The input interface includes the following parameters:

- List of calculations to be checked.
- List of calculations on which the models should be trained.
- List of settings for check functions, like thresholds or model settings.

The output of check functions is a list of validation messages, the structure of which is explained in the next section. Thus such checks can be flexibly combined to fit to the company specifics, independent of the used methods and underlying models.

Though very different models and methods can be used for plausibility checks, in general they consist of the following phases:

(1) Training models: In this phase, we analyze the historical data of similar products to automatically extract the typical values and (sub-) structures for different aspects of the costing structures, as well as typical variations of them.
(2) Detecting anomalies: During this phase, the potential errors are detected as anomalies and their impact is calculated.

The product calculations have hierarchical structures, which consist of items and modules of a product with such properties like prices, quantity, process duration, maturity, etc. Accordingly, the plausibility checks use different models that evaluate different aspects of those structures. Depending on the used models, the plausibility checks can be classified into the following groups:

- Scalar value checks: These check if the scalar values like prices or duration of manufacturing processes significantly deviate from the typical values.
- Structure checks: These check if the current structures deviate from the typical pattern structures of the similar products. For example, one check detects if some item is missing in the sub-module of the product, whereas it is always present in similar sub-modules from the training data-set.
- Cost share checks: Due to different product designs, the shares of different types of costs like ratio between material costs and processing costs may get unacceptable. This will indicate the general structural problems within the product.

3 PLAUSIBILITY CHECK RESULTS

The model training and anomaly detection in plausibility checks can be realized by different data analysis and machine learning methods, from basic statistical approaches, to classical machine learning methods all the way to deep learning.

When using the statistical approach for the scalar value checks, the statistical indicators like average, median and standard deviation are at first calculated from the training data. The anomalies are then detected using the variance test, which indicates the value deviation by more than 1.5*standard deviation from the average value. The statistical approaches are working well in case of small to medium amount of training data, allowing for a quick training and detecting plenty of anomalies which would be otherwise very hard to identify manually.

If more complex dependencies are available in the data, the more complex approaches are helpful. For example, to model the dependency of times necessary for processing of the subparts on different parameters of the final product we have used the Support Vector Machines, which work well in case of non-linear dependencies. For classification and prediction of the substructures, the recurrent neural networks have been used.

We have trained a set of plausibility check models on the data-sets from 4 customers representing different industries like original equipment manufacturers (OEMs), machine-tool producers and automotive suppliers. Notably the data has shown a great variety in the structure and underlying models, having from 1-2 up to several

Table 1. Result of a plausibility check with different types of plausibility check messages.

LINE	ITEM	MSG TYPE	MESSAGE TEXT	COST IMPACT
1	100-110 Slug for casing	10	Price (variable portion) of 20.0 EUR differs from usual one of 10.0 (variance of 0.6)	10.0 EUR
2	100-110 Slug for casing	11	Price (fixed portion) of 1.0 EUR differs from usual one of 0.0 (variance of 0.0)	1.0 EUR
3	AT2 pick according to pick list	15	Duration of 30.0 MIN differs from usual one of 16.0 (variance of 4.82)	528.0 EUR
4	AT2 inspect and deliver to storage	16	Duration of 10.0 MIN under item '100-300 Shaft' differs from usual one of 5.00 (variance of 0.0)	5.40 EUR
5	NO ITEM	22	The item 'AT2 pick according to pick list' is present 3 times, usual is 2.0 times (variance of 0.0)	765.0 EUR
6	NO ITEM	26	The item 'AT1 clamp impeller (setup)' is missing under assembly '100-200 Drive' (normal probability of 1.00)	7.20 EUR
7	100-120 Flat seal	51	Maturity: Item was last modified 170.68 days before last calculation version update and may not be up-to-date	110.0 EUR
8	Calculation Version	90	Cost component '110 (Materials (AG 110))' has share of 13.63% which differs from usual one of 14.86% (variance of 0.73%)	-127.0 EUR
9	Calculation Version	90	Cost component '120 (Activities (AG 120))' has share of 63.09% which differs from usual one of 61.93% (variance of 0.67%)	119.62 EUR
10	Calculation Version	91	Calculation version has total cost of 9649.67 EUR which differs from usual one of 8154.61 EUR (variance of 128.41 EUR)	149.05 EUR

hundreds calculations in the project or from 10 up to 50000 items in a calculations. The results of training and plausibility checks have been validated together with customers. An example of the resulting list of plausibility check messages is shown in Table 1.

The resulting messages contain the following fields:
ITEM: indicates which item or submodule of the costing structure the message refers. Some messages refer to a specific item, and the others to the whole calculation version.
MESSAGE TYPE: indicates the message type. Each plausibility check method can produce at least one message type. The unique message types allow the quick overview and filtering over the messages.
MESSAGE TEXT: contains the detailed description of the identified issues, including the problematic field, and the current value of it. Further, it contains the typical values and variations of the values for this kind of item, which allows to follow why the plausibility check message was fired. Moreover, the user can then see how far the current value is from the expected one and thus justify it, so that the system can learn from his feedback.
COST IMPACT: presents which sum of the total cost may be potentially affected by the issue. This allows the user to prioritize the issues and address the most critical ones at first.

All these details help the user to follow the causes of the identified issues and to make the decisions on how to correct them. For example, in Table 1 the item 'AT2 pick according to pick list' has the most critical cost impact with sums of 528.0 EUR and 765.0 EUR and thus should be considered at first. Further, line 3 gives a hint that the processing duration is with 30 minutes unusually long and may be corrected towards 16 minutes. In line 5, this processing step is present 3 times which is may be 1 time too much. The impact on the cost shares and total costs

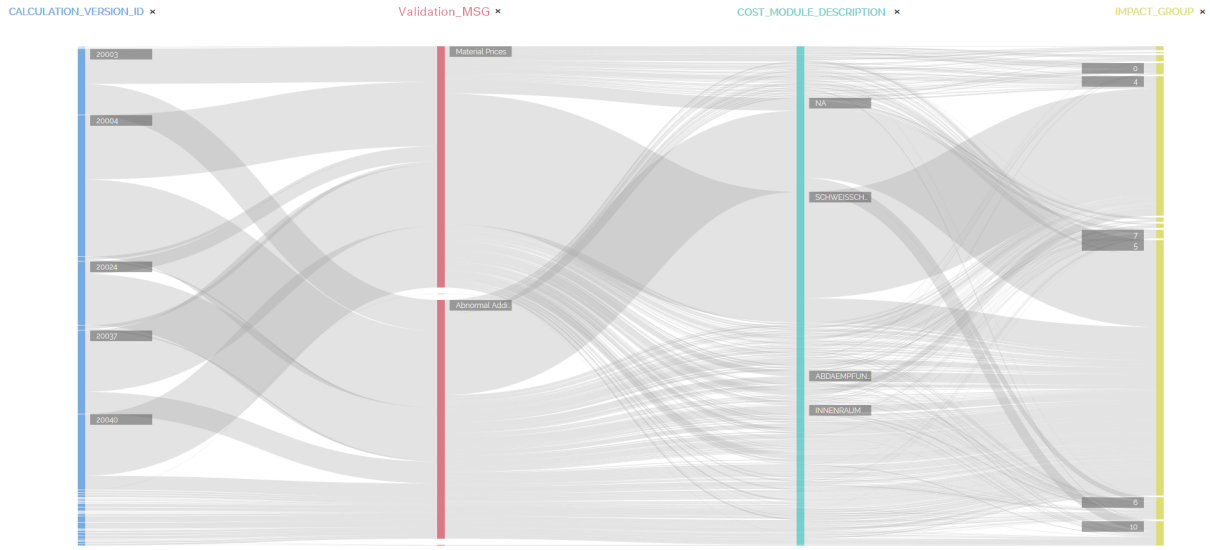

Fig. 1. Parallel Sets visualization of the result of our machine learning method on product costing structure with 92 items.

indicated in lines 8, 9 and 10 are subsequent deviations caused by anomalies on the item levels, giving a hint that the share of materials in the whole calculation is too small due to wrong entries on processing steps.

Above all, the plausibility checks only give hints on the deviations from the typical values and the messages do not necessarily indicate the errors. In some cases, such deviations are intended for new products and the user can accept them. Furthermore, some kinds of checks may require the adaptation to the company specifics.

Depending on the calculation size, quality of the data and the used plausibility checks, the number of validation messages can vary from 0 to several thousand. Presenting a plain flat list of validation messages would overwhelm the user and hinder his work. First, because of the sheer number of messages, and second, because of the large number of dimensions (20) associated with each message. To help the user to make sense of this deluge of data, a novel interactive visualization tool is required.

4 VISUAL EXPLORATION OF MACHINE LEARNING RESULTS

Recent works have shown that applying novel visualization techniques can help to understand PLC data faster and easier [Vosough et al. 2017a,b]. However, there is currently no approach designed to analyze multidimensional categorical data resulting from ML-based plausibility checking algorithms as currently implemented in SAP PLC application – as explained previously in section 3.

4.1 Choice of Visualization Technique

A considerable number of advanced visualization techniques has been proposed for representing multidimensional data and many of them are reviewed by de Oliveira and Levkowitz [De Oliveira and Levkowitz 2003]. Most of these visualization techniques aim to display more than two data dimensions, and facilitate the data interpretation for users. When considering the suggested visualization techniques, the question is: which one fits better to the type of data obtained from our machine learning algorithm. To that end, the number of dimensions, number of

Fig. 2. Parallel Sets visualization of the result of our machine learning method on product costing structure .

variates, number of data items, and data types were considered. In addition, the tasks carried on by users was defined with the customers as co-innovators of the project.

Several visualization solutions already exist to analyzing large multidimensional data-sets. Well known examples include Scatterplot Matrices [Andrews 1972], Parallel coordinates [Inselberg and Dimsdale 1987] or TreeMaps [Tufte 1985]. Keim has defined a classification for multidimensional visualization techniques, drawing a table that compares the available techniques [Keim 2000]. Based on this classification, Parallel Coordinates and related geometric techniques emerge as a common solution for representing a large number of dimensions but with few data items per dimension.

Parallel Sets is another obvious choice for visualizing multidimensional categorical data. Parallel Sets was the first parallel coordinates variant to utilize ribbons to represent data-subsets as a whole instead of drawing multiple individual polylines [Bendix et al. 2005; Kosara et al. 2006]. Considering the task taxonomy in Alsallakh et al. [Alsallakh et al. 2014], we selected Parallel Sets to represents the results of our ML-based PLC plausibility checks as it can support exploration of the relationship between multiple dimensions.

In addition to the visualization techniques, interaction techniques can play an important role for effective data exploration. Therefore, some interaction methods were added to the visualization solution such as selecting and highlighting specific data items and the possibility to get more information via tool-tip messages.

4.2 Applying the Visualization

In the following, we look at two realistic data-sets from an industrial machine and automotive industry. Figure 1 represents the validation messages found in one of the SAP customers data-sets. We applied our ML-based PLC plausibility check algorithms to this data-set and found 1911 potential errors. Beside the validation errors, the result contains 11 individual data properties that plausibility results span. In Figure 1, 4 of the categorical dimensions are

shown and the number of validation messages represents the data quantity which is the thickness of the ribbons. Each vertical axis represents an individual data property. The first dimension is the *Calculation_Version_Id* which is shown by the first axis (blue). Typical PLC projects have several calculation versions created over the course of the project's lifetime. The second dimension is the *Validation Message* (Validation_MSG) shown in pink. The validation message represents different message types returned by our plausibility check algorithm. The third dimension is the *Cost_Module_Description* shown in green. This dimension indicates which product modules these validation messages belong to. Furthermore, different error messages have different impacts on the product's total cost. Those impacts are categorized in 10 different categories and show on the *Impact_Group* dimension (yellow).

To follow the validation results, a possibility of deep dive into the values is provided. The users can get the exact and absolute values of an item using mouse-over tool-tips. In case of material price checks, the users can see the problematic actual value, as well as the average value and the standard deviation for training data in a candle diagram [Morris 2006]. This allows to understand how far is the detected value from the typical ones. Further, this gives the hint on a possible countermeasure, e.g. to set the new value within the interval of typical values. In this way the important task of cost optimization support [Walter et al. 2018] can be implemented.

The second example shown in Figure 2 shows the 261 error messages found in another customer's data-set. The data has 13 dimensions, and among those we show 3 dimensions in Figure2. The thickness of ribbons represents the cost impact in this example. Different validation messages (pink) are shown along with different calculations (blue) and corresponding calculation versions (green). In this example, we see immediately that the cost impact of the selected calculation – shown in the picture – is mainly caused by two validation messages. The first message is on "Material Prices", indicating that this item's price differs from usual one. The second message refers to the "Abnormal Addition Item" validation message, which happens when an item is not expected to present in such calculation.

The first drop down on top of the screen (left) is used to add new dimensions, the second one (middle) to change the data quantity and the third one (right) to change the data-sets. Axes can be manually rearranged by dragging, or removed by the small cross symbol placed on the right side of their names that appears in red color after hovering the mouse over a dimension's area. The colors of the axes follow Paul Tol's categorical color scheme Palette II [Tol 2012].

The figures give an overview of which validation messages have been found for each module. They help to quickly detect the problematic areas and assign blame to the responsible contributor. Also, when selecting an item in one dimension, all connected items and relevant ribbons are highlighted.

5 SUMMARY AND OUTLOOK

In this article, we presented a Parallel Sets based visualization solution to show the results of plausibility checks in product costing applications. We used machine learning techniques to process and visualize large data-sets of two SAP Product Lifecycle Costing customers.

Our plans for future work is to offer an engineered service by consulting the customers of the project. Apart from applying the machine learning algorithm to other customers' data-sets, we are planning to extend our visualization technique to cover more aspects of the data -like the hierarchical feature- in the future. Moreover, the visualization will be evaluated with the customers of the project during one of the upcoming customer's workshop.

In addition, one limitation of the currently used visualization is that we can only show one quantity at the time. Another interactive feature, which would be highly beneficial to add, is the possibility to visualize different quantities at the same time for better comparison and easier decision making.

ACKNOWLEDGMENTS

The authors would like to thank all members and customers of SAP Product Lifecycle Costing for their input on different parts of this research, and special thanks to Marius Hograefer for implementing the Parallel Sets prototype.

REFERENCES

Bilal Alsallakh, Luana Micallef, Wolfgang Aigner, Helwig Hauser, Silvia Miksch, and Peter Rodgers. 2014. Visualizing sets and set-typed data: State-of-the-art and future challenges. In *Eurographics conference on Visualization (EuroVis)–State of The Art Reports*. 1–21.

David F Andrews. 1972. Plots of high-dimensional data. *Biometrics* (1972), 125–136.

Fabian Bendix, Robert Kosara, and Helwig Hauser. 2005. Parallel Sets: Visual analysis of categorical data. In *Proc. of the IEEE Symposium on Information Visualization (InfoVis'05)*. IEEE, 133–140. https://doi.org/10.1109/INFVIS.2005.1532139

Indranil Bose and Radha K Mahapatra. 2001. Business data miningâĂŤa machine learning perspective. *Information & management* 39, 3 (2001), 211–225.

Hsinchun Chen, Roger HL Chiang, and Veda C Storey. 2012. Business intelligence and analytics: from big data to big impact. *MIS quarterly* (2012), 1165–1188.

MC Ferreira De Oliveira and Haim Levkowitz. 2003. From visual data exploration to visual data mining: a survey. *IEEE Transactions on Visualization and Computer Graphics* 9, 3 (2003), 378–394.

Alfred Inselberg and Bernard Dimsdale. 1987. Parallel coordinates for visualizing multi-dimensional geometry. In *Computer Graphics 1987*. Springer, 25–44.

Daniel A. Keim. 2000. Designing Pixel-Oriented Visualization Techniques: Theory and Applications. *IEEE Trans. on Visualization and Computer Graphics* 6, 1 (Jan. 2000), 59–78. https://doi.org/10.1109/2945.841121

Robert Kosara, Fabian Bendix, and Helwig Hauser. 2006. Parallel Sets: Interactive exploration and visual analysis of categorical data. *IEEE Transactions on Visualization and Computer Graphics* 12, 4 (2006), 558–568. https://doi.org/10.1109/TVCG.2006.76

Greg L Morris. 2006. *Candlestick Charting Explained: Timeless Techniques for Trading Stocks and Futures: Timeless Techniques for Trading stocks and Sutures*. McGraw Hill Professional.

Paul Tol. 2012. *Colour Schemes*. Technical Report SRON/EPS/TN/09-002 v.2.2. SRON Netherlands Institute for Space Research.

Edward R Tufte. 1985. The visual display of quantitative information. *Journal for Healthcare Quality* 7, 3 (1985), 15.

Zana Vosough, Rainer Groh, and Hans-Jörg Schulz. 2017a. On Establishing Visualization Requirements: A Case Study in Product Costing. In *Eurographics Conference on Visualization (EuroVis) : Short Papers*. The Eurographics Association, to appear.

Zana Vosough, Dietrich Kammer, Mandy Keck, and Rainer Groh. 2017b. Visualizing Uncertainty in Flow Diagrams: A Case Study in Product Costing. In *Proc. of the International Symposium on Visual Information Communication and Interaction (VINCI'17)*. 1–8. https://doi.org/10.1145/3105971.3105972

Zana Vosough, Matthias Walther, Jochen Rode, Stefan Hesse, and Rainer Groh. 2016. Having Fun with Customers: Lessons Learned From an Agile Development of a Business Software. In *Stakeholder Involvement in Agile Development - Workshop at ACM NordiCHI 2016* (24. October) *(NordiChi)*. ACM.

Matthias Walter, Christian Leyh, and Susanne Strahringer. 2018. Toward Early Product Cost Optimization: Requirements for an Integrated Measure Management Approach. In *Proceedings of Multiconference Wirtschaftsinformatik 2018 (MKWI2018). Band V: Data driven X âĂŤ Turning Data into Value*. Leuphana University Lueneburg, 2057–2068.

Ian H. Witten, Eibe Frank, and Mark A. Hall. 2016. *Data Mining: Practical Machine Learning Tools and Techniques* (4 ed.). Morgan Kaufmann, Amsterdam.