IFC-Bank of Italy Workshop on "Data Science in Central Banking: Applications and tools"

14-17 February 2022

Interactive visualization tool: outlier detection in large multidimensional datasets[1]

Christoph Leitner, Thomas Kemetmueller and Philipp Reisinger,
National Bank of the Republic of Austria

[1] This contribution was prepared for the workshop. The views expressed are those of the authors and do not necessarily reflect the views of the Bank of Italy, the BIS, the IFC or the other central banks and institutions represented at the event.

Interactive Visualization Tool

Outlier detection in large multidimensional datasets

Philipp Reisinger, Thomas Kemetmüller, Christoph Leitner[1]

Abstract

The Oesterreichische Nationalbank (OeNB) collects granular credit data from several hundred domestic reporting agents (banks and financial institutions) in a Granular Credit Register (GCR) and in an integrated manner with the ESCB's AnaCredit. Per reporting date, the GCR contains approximately 1.2 million credit instruments with well beyond one hundred data dimensions. To assure data quality in such complex granular datasets, powerful approaches and tools are needed. At OeNB a variety of statistical methods (including statistical inference, regression models and outlier detection algorithms) is used to identify outliers and potentially incorrect values.

This paper presents visualisation techniques to support these statistical methods by additional results as well as simplifying these results by appropriate plots. The focus here are parallel coordinates plots which allow the visualization of multidimensional granular datasets but require a high level of interactivity. Hence, we developed a browser-based interactive visualization tool to appropriately utilize parallel coordinate plots and other techniques to explore large datasets. An example for the GCR demonstrates the usage of this tool to generate insights, identify anomalies, and thus helps to improve data quality.

Keywords: Data visualization, parallel coordinates, outlier detection

[1] Oesterreichische Nationalbank – Supervisory Statistics, Models and Credit Quality Assessment Division

1. Introduction

Under the ECB Regulation on AnaCredit (see EZB VO (EU) 2016) as well as the national banking legislation on granular credit data (see GKE-V 2018 and § 75 BWG 2022) the Oesterreichische Nationalbank (OeNB) collects monthly credit data from several hundred domestic reporting agents (banks and financial institutions) on a granular basis. The integrated data (implemented employing an entity relationship model) forms the henceforth called Granular Credit Register (GCR). The GCR contains approximately 1.2 million credit instruments with well beyond one hundred numerical and categorical data dimensions per reporting date.

Reported data must fulfil an extensive set of validation rules and plausibility checks upon transmission by reporting agents to avoid the acceptance of faulty data. The distinction between validation rules and plausibility checks is that the former identify unambiguous reporting errors with certainty, while the latter identify data that are likely to be incorrect. Still, accepted data may not necessarily be fully correct. Therefore, downstream quality assurance is required to improve overall quality of the data. In this step OeNB uses statistical methods like statistical inference, regression models and other effective outlier detection algorithms.

Following Anscombe (1973), who showed that summary statistics can obscure outliers, which could have been easily spotted by plotting the data, we experimented with visual methods to utilize human pattern recognition abilities. In the end, we developed a customized tool ("PARVIZ") to support outlier detection in the GCR. Chapter 2 describes the chart types incorporated in PARVIZ. Chapter 3 explains its design, set up and features. Chapter 4 demonstrates features of the tool in a use case identifying outlier credit instruments in the GCR.

2. Methods

In the context of analysing datasets (e.g., identifying outliers) the distribution of dimensions and the correlation between dimensions play an important role. There are a lot of possibilities to do this using visualization. The need to work with highly dimensional granular data limits this set. Parallel coordinates plots (PCPs) can show several data dimensions on parallel axes and are capable of handling large granular datasets but are intended to be used with numerical dimensions (see Figure 1).

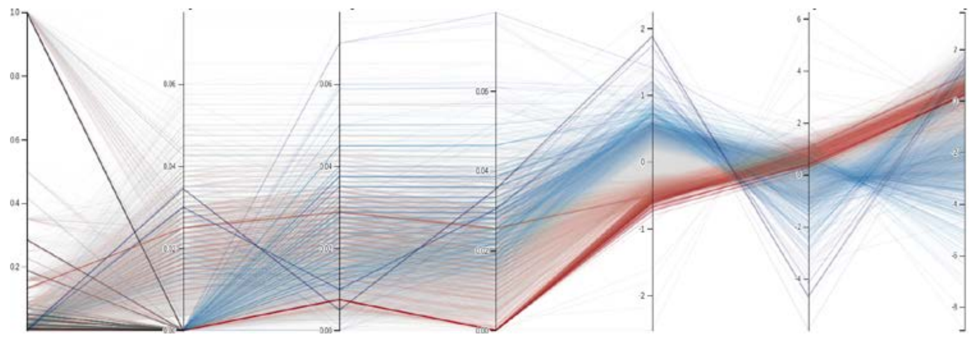


Figure 1: An example of parallel coordinates plot displaying data with seven numerical dimensions. Each observation is represented by a polyline, cutting the axes at the observation's value in the corresponding dimension. The colour of the polylines indicates z-scores of the 5th dimension (> 0.6745: blue; < -0.6745: red).

Other chart types, like parallel sets (see Kosara *et al,* 2006) or mosaic plots (see Hartigan and Kleiner, 1984), are better for categorical data, but were omitted due to the involved aggregation of values per category, lacking the required granularity. The PCPs shortcoming with categorical dimensions can be compensated by displaying aggregated values as bar charts for each respective axes, indicating the dimensions' distribution much like histograms. Sometimes box plots are used to show the distribution of numerical variables but since they only rely on simple summary statistics (and share the potential flaw presented by Anscombe, 1973), we chose violin plots to explicitly display the probability density. PCPs indicate the correlation of adjacent axes. However, Li *et al* (2010) showed that scatter plots are better for visual correlation analysis. As Figure 2 shows that patterns can only be identified between adjacent axes in PCPs, it would be useful to give users the possibility to easily modify axes order (along with their rotation and scaling). An interactive implementation of such a plot can solve this issue.

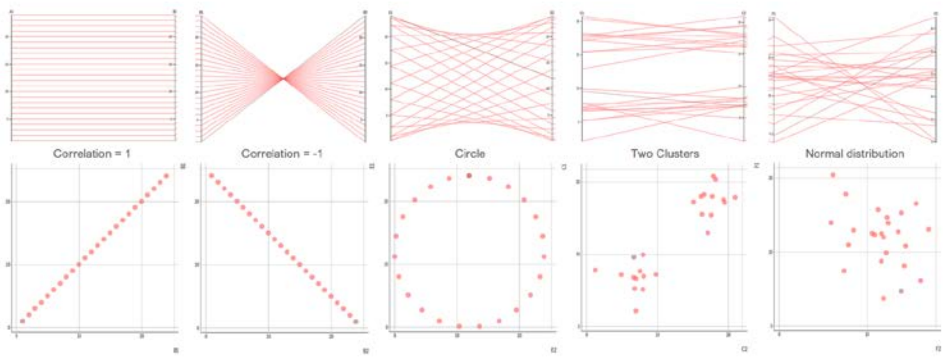


Figure 2: Parallel coordinates plots and scatter plots can be used to evaluate correlation in different data examples.

3. The interactive visualization tool PARVIZ

The interactive visualization tool (called "PARVIZ") is designed as pure front-end web application, implemented with standard web technologies[2] and additional libraries: D3.js (Bostock, 2018), d3.parcoords.js (Chang, 2016 and Xing, 2019), d3.tip (Palmer, 2013), SlickGrid (Leibman, 2012), underscore.math (Chang, 2012), jQuery (jQueryTeam, 2016), jQueryUI (jQueryTeam, 2011), underscore.js (Ashkenas, 2012). The charts in PARVIZ are based on D3.js-examples by Bostock (2021), Davis (2021), Chang (2021), Petersson (2018), Galavotti (2017) and Holtz (2018).

The tools' functionalities are contained in a single file (PARVIZ.html) that runs locally in a web browser (no backend). PARVIZ does not require an internet connection – all necessary libraries are included in the HTML-file (with only 1 MB). Data to be displayed is stored separately as JSON[3]-file in a secured folder on a network drive, preventing unauthorized access. The path to the secured data is contained in an additional JSON-file (2 KB), which allows to switch between different datasets without editing the main HTML-file. The specified dataset is then loaded when opening the HTML-file.

The initial view of the user interface contains a menu bar on the top, a PCP in the centre and a table at the bottom. All dimensions of the loaded data are included in the table, whereas only the selected (via menu) are displayed in the PCP. For each dimension with numerical values the distribution is shown using a violin plot on the axis. For each categorical dimensions a bar chart is shown on the axis. These axis plots can be hidden by changing their opacity in the menu bar. Missing/null values are displayed below the horizontal line at the bottom of the PCP.

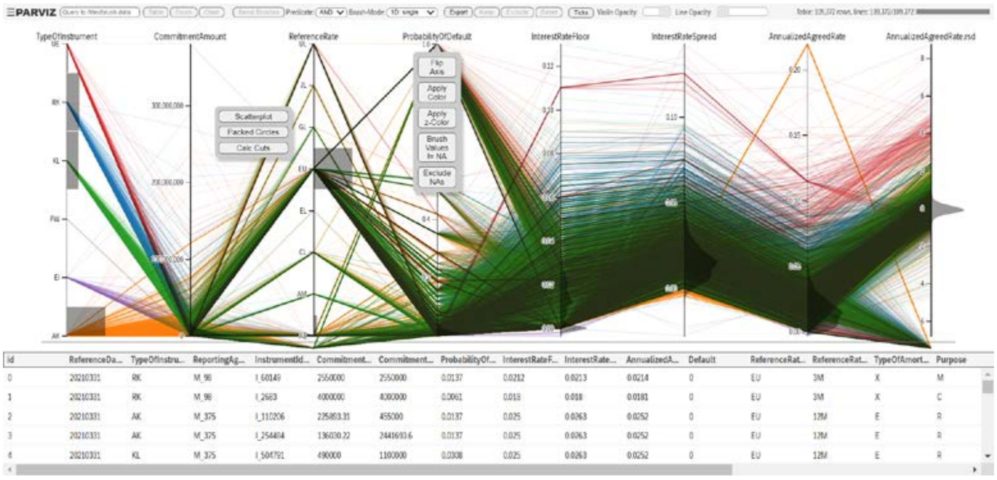


Figure 3: PARVIZ' user interface displays here two categorical and six numerical dimensions as parallel coordinates plot and a table with all dimensions of the corresponding dataset.

On hovering over table rows, the corresponding polyline is highlighted in the PCP (for persistent highlighting multiple rows can be "left-clicked"). Selecting and highlighting polylines is especially important when many polylines are plotted and individual polylines cannot be distinguished ("overplotting"). Brushes allow to select polyline bundles based on value ranges per axis ("single": one range per axis, "multi":

[2] JavaScript, CSS, HTML, SVG, Canvas

[3] JavaSript Object Notation

multiple ranges per axis) or characteristics of lines between two adjacent axes ("strums": lines intersecting with a given strum, "angular": lines sloping within a given range specified by a circle segment). Figure 4 shows range-based brushes on the left (top: "single", bottom: "multi") and line-based brushes on the right (top: "strums", bottom: "angular"). Multiple brushes (of one type) can be joined using a logical "AND" or "OR" to allow for comprehensive data queries, filtering the tables rows accordingly.

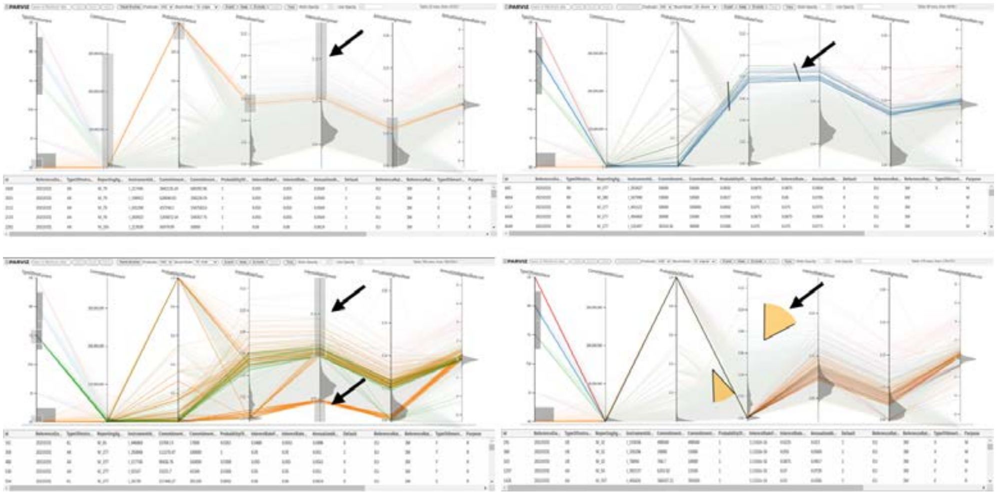


Figure 4: Different brushing methods (top left: "single", bottom left: "multi", top right: "strums", bottom right: "angular") allow targeted selections.

In addition to brushing the PCP via mouse interaction, an input box is available for a more precise selection and can be used to brush the plot or filter the table. The count of all currently selected rows and lines is displayed in the menu bar. For further reference, all rows of the selection are easily exportable in a CSV-format by a single click. Selections can be kept or excluded, triggering the rescaling of all axes, which is useful to zoom in on certain dimensions. Axes are reorderable (drag-and-drop) and invertible via an axis menu. Double-clicking on an axis header shows the axis' menu, which also enables users to quickly brush or exclude missing values and change the polyline colour to express the respective dimension – for categorical dimensions distinct colours are applied and with numerical dimension a colour gradient is used to differentiate high and low values or the z-scores[4] of these values. When clicking on a column header of a numerical dimension in the table, a box plots is generated. By clicking on an element of the box plot the brushing in the PCP can be controlled (see Figure 5).

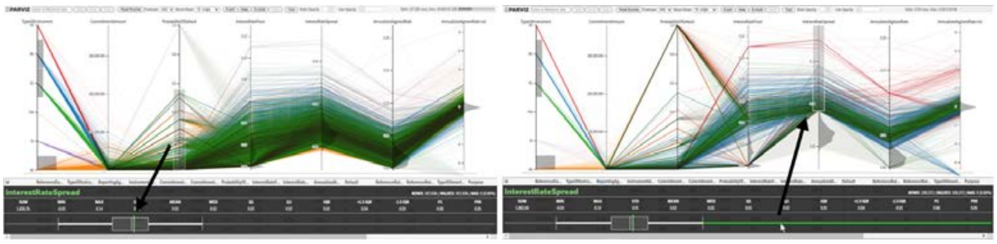


Figure 5: By clicking on a column header of numerical dimensions displays a box plot for all selected polylines (left). By clicking on an element of the box plot the brushing in the PCP can be controlled (right).

[4] The z-score of a value is the standardized value, computed via (value – mean of all values of the same dimension) / standard deviation of all values of the same dimension.

Scatter plots are displayed by clicking between two adjacent axes. A tooltip shows configurable info for each dot (see Figure 6). Clicking on a dot filters the respective row in the data table. By hovering this row, the corresponding polyline in the PCP is highlighted. Packed circles show how a numerical dimension is distributed across multiple levels of categorical dimensions (see Figure 7). By clicking on the right of a numerical axis, its values are hierarchically aggregated for all categorical dimensions to the right of this numerical dimension, enabling users to quickly configure the plot by arranging the axes accordingly. Especially for the purpose of outlier detection, we also included a feature to identify polylines that diverge from the majority by showing the number of all other polylines that each polyline cuts (as total over all axes as well as for each adjacent axes-pair) on additional axes in the PCP. In this sense, the number of polyline cuts can be regarded as an outlier score where the polylines that "buck the trend" are potentially outliers and should be further investigated.

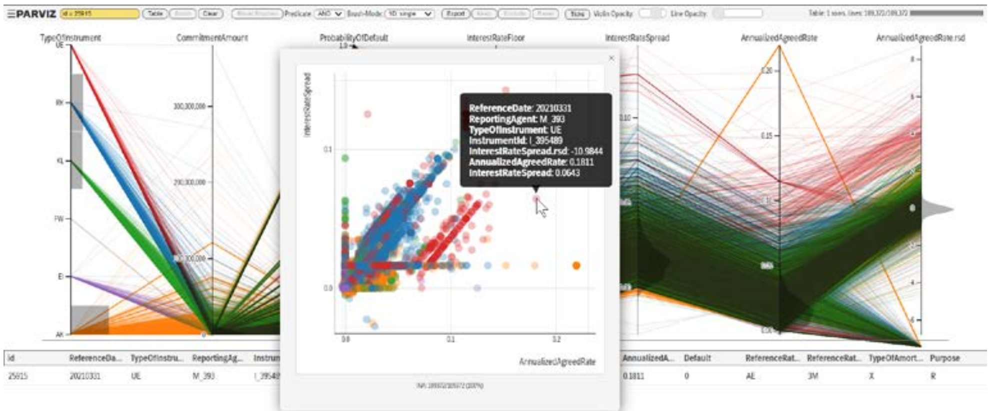


Figure 6: By clicking between two adjacent axes, a zoomable scatter plot (with tooltip) is displayed for those dimensions.

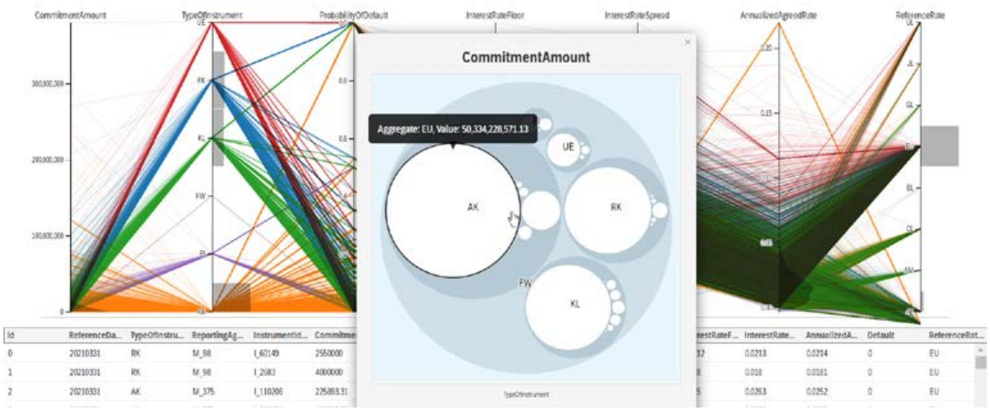


Figure 7: Zoomable packed circles plots of a numerical dimension aggregated over categorical dimensions (here "TypeOfInstrument" and "ReferenceRate") help to explore categorical dimensions of datasets.

4. Using PARVIZ to identify outliers in the GCR

In this chapter we use a sample of the GCR to demonstrate features of the visualization tool with the goal to identify outlier credit instruments in the data. The data contains 109,372 rows (credit instruments) and 20 dimensions of one reporting date and is anonymized for this paper (see Table 1). The last three of those dimensions (indicated by the extension ".rsd") depict outlier scores. These scores are based on the residuals of a linear model (measured in multiples of the standard deviation of the residuals distribution) predicting the respective dimension using other dimensions as the explanatory variables. High absolute residuals reflect potential outliers (for more information on this method see Aggarwal, 2017).

Dimensions

Table 1

Name	Type	Units	Example
ReferenceDate	categorical	-	"20210331"
TypeOfInstrument	categorical	-	"AK"
ReportingAgent	categorical	-	"M_12"
InstrumentId	categorical	-	"I_12345"
CommitmentAmount	numerical	€	12,345.67
CommitmentAmountAtInception	numerical	€	12,345.67
ProbabilityOfDefault	numerical	%	0.2345 (=23,45%)
InterestRateFloor	numerical	%	0.02 (=2%)
InterestRateSpread	numerical	%	0.02 (=2%)
AnnualizedAgreedRate	numerical	%	0.02 (=2%)
Default	numerical	0/1	1 (=Default)
ReferenceRate	categorical	-	"EU"
ReferenceRateMaturity	categorical	-	"3M"
TypeOfAmortization	categorical	-	"X"
Purpose	categorical	-	"R"
CarryingAmount	numerical	€	12,345.67
ImpairmentAmount	numerical	€	12,345.67
AnnualizedAgreedRate.rsd	numerical	σ	1.2345
InterestRateSpread.rsd	numerical	σ	1.2345
InterestRateFloor.rsd	numerical	σ	1.2345

Extreme values are natural candidates for potential outliers. In PARVIZ they are easily identified by examining polylines on each axis of the PCP and brushing relevant ranges. Brushing several axes quickly reveals local extreme values in subsets (cf. Figure 8).

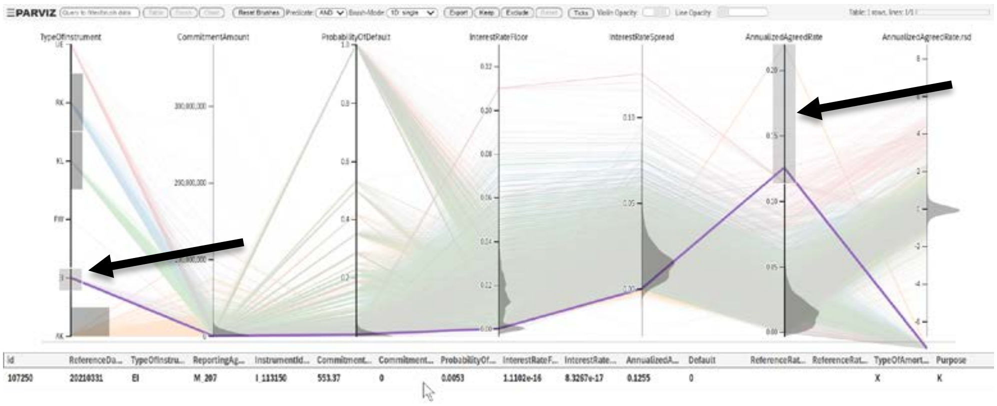


Figure 8: The maximum value of dimension "AnnualizedAgreedRate" for one credit instrument type (dimension "TypeOfInstrument" = "EI") can be easily found by brushing two axes in the parallel coordinates plot.

The polyline of the highlighted instrument (id = 107250) in Figure 8 shows – despite having the highest annualized agreed rate of all instruments of the same type ("EI") – that

(1) the interest rate spread is very low, resulting in a line that goes against the flow of most other lines between the adjacent axes;

(2) the instrument was not included in the outlier model for "AnnualizedAgreedRate" (indicated by the missing value on the last axis).

To further investigate the apparent correlation between the interest rate spread and the annualized agreed rate (indicated by the fact that most lines are parallel), a scatter plot for all credit instruments of this type is generated. The scatter plot confirms the relationship and detects a few outliers (including credit instrument with id = 107250; see Figure 9). Using the tools' feature to count the intersections of each line with all other lines also identifies this credit instrument (see Figure 10).

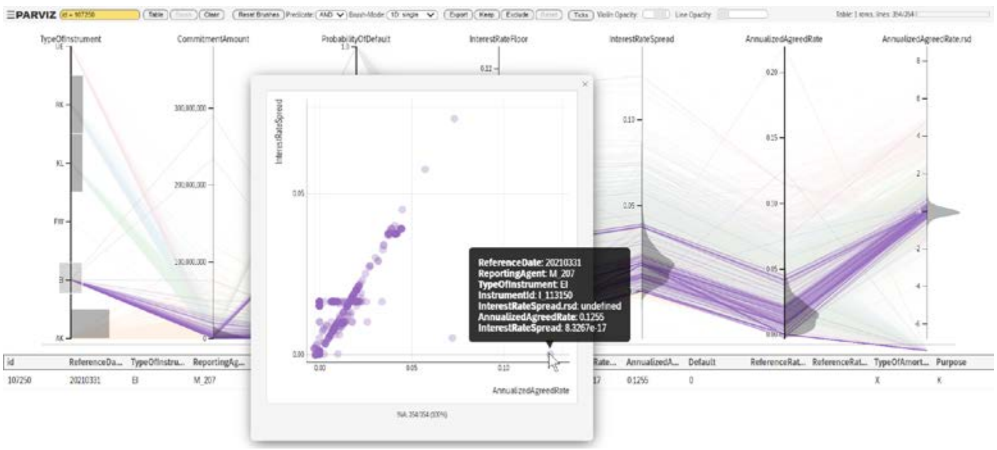


Figure 9: The scatter plot confirms the relationship and detects the instrument with the id = 107250 as outlier.

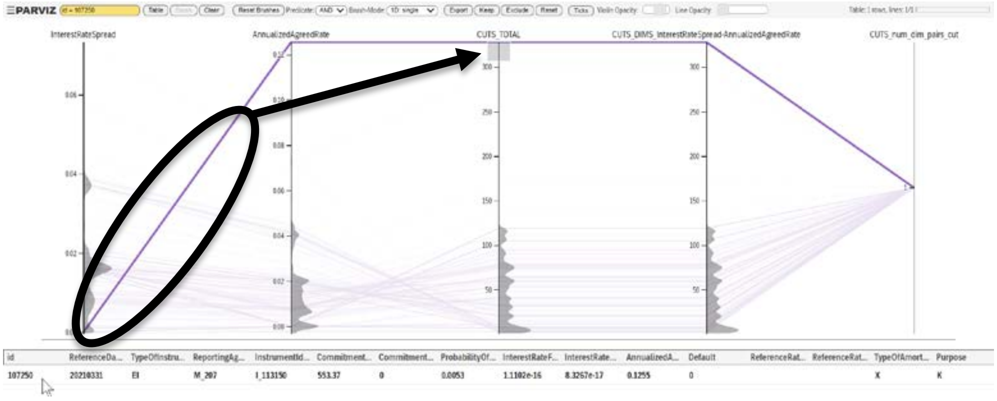


Figure 10: Counting the number of intersections of each line with all other lines shows the instrument with id 107250 as outlier (cutting far more lines than any other line/credit instrument).

The interactive box plot of the respective table column helps to verify the visually assessed anomaly of the inspected credit instrument. Selecting the box plots' upper outliers (> upper whisker) of the dimension "AnnualizedAgreedRate" and all lower outliers (< lower whisker) of the dimension "InterestRateSpread" results in two credit instruments: The known instrument with id = 107250 (see above) and another one (see Figure 11). The high residuals strengthen the finding that the combination of high annualized rates and low interest rate spreads are being suspicious. These residuals provide valuable guidelines not only for validating outliers but also to find them: By scrutinizing the lines slope in Figure 12, it seems that high residuals are connected to this unusual combination of interest rate attributes for all instrument types. This presumed relation can be exploited by using the angular brush to select all lines (including those with missing values on the residuals axis) with a similar slope to expand the models results to credit instruments not assessed by the model. Selecting all upward sloping lines of credit instruments with type "EI" leads to the same result as above. This time only by interpreting and analysing model results.

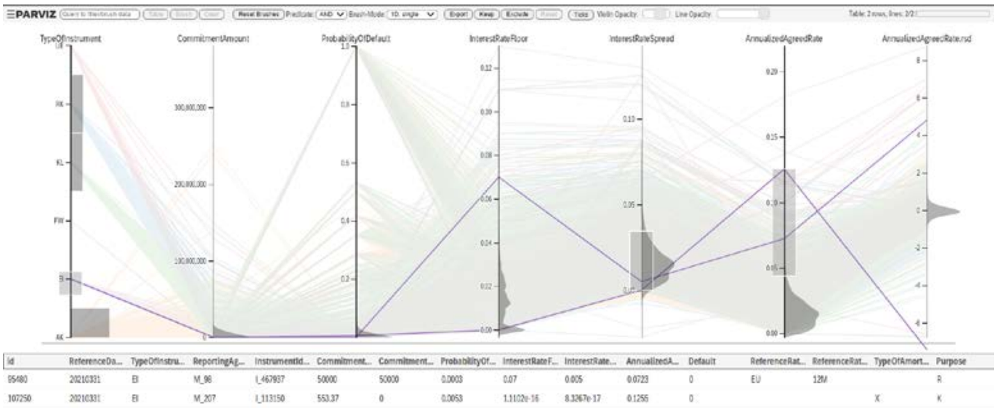


Figure 11: Two credit instruments identified as outliers by the combination of being an upper outlier (> upper whisker) of the dimension "AnnualizedAgreedRate" and a lower outlier (< lower whisker) of the dimension "InterestRateSpread".

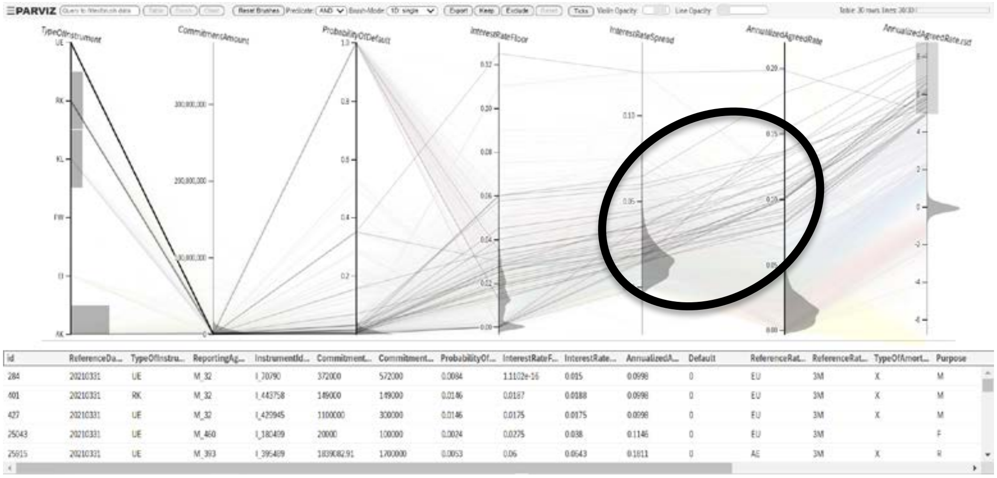


Figure 12: The high residuals on the last axis are apparently connected to the highlighted upward sloping lines.

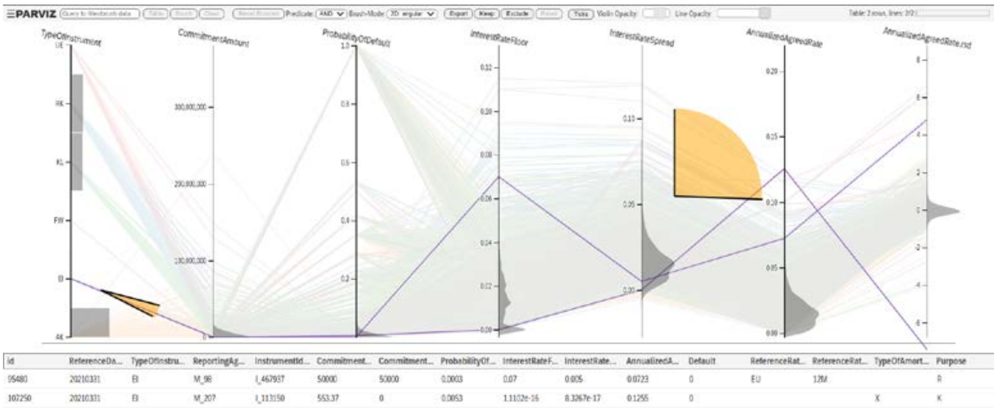


Figure 13: Selecting all upward sloping credit instruments (of type "EI") to identify the same two credit instruments as displayed in Figure 11.

Discussion and Conclusion

Visualizations can improve the understanding of large datasets by enabling users to quickly grasp the underlying data's structure. In addition, a selection of appropriate chart types combined with a fast interactive user interface helps identify potential anomalies in large datasets and subsequently increase data quality. Even though the results obtained from visually inspection and outlier detection do not guarantee that specific datasets are incorrect or erroneous data, they can justify further investigation. From our experience, derived indicators like outlier scores obtained from statistical modelling support the identification of outliers. PARVIZ combines all these aspects in one interactive visualization tool.

Interactive visualizations of big data can place considerable demands on computational resources. Since our tool runs in the browser on the user's machine, its performance is limited by local resources. As a consequence, loading large datasets and rendering all polylines of the PCP may take considerable time (several minutes). The data is only loaded once when PARVIZ is started, but polylines must be rendered frequently (e.g., on reordering or inversion of axes).

From our point of view, following the strategies to minimize the rendering duration of the PCP of the parcoords' library (see Chang, 2016 and Xing, 2019), like

- the use of Canvas to draw polylines (not of SVG),
- progressive rendering (to allow for interactions while lines are still being rendered),
- rendering prioritization (to render brushed polylines first),
- background rendering (to continue rendering after all brushed polylines are rendered);

is useful.

Additionally, the library allows to configure the render rate (number of polylines drawn at once). The higher the render rate, the faster the rendering process, but the greater the lag of user interactions. Lower render rates allow for more fluid user interactions but take longer to complete the plot. Concerning this trade off, we prefer to not inhibit users in their actions and opt for a low render rate (e.g., 100).

The design of our tool does not allow the use of larger decentralized server resources. Even the use of a powerful backend could possibly not alleviate the problem due to the high degree of interactivity that determines the plots appearance. For the time being, additional performance improvements only seem possible by increasing local resources and further optimizing the current code.

Performance requirements naturally increase with the size of data. So, we advise selecting useful subsets to reduce the computational burden and avoid overplotting. We recommend restricting plotting to only the data required for the current task. With advancements of the capabilities of modern web browsers and desktop computers performance will be further improved, allowing to handle increasing amounts of data.

Readily available and well documented web development techniques combined with open-source frameworks and a vast selection of visualization examples facilitate the development of tailor-made solutions to interact with data. We highly encourage to take advantage of these possibilities and create customized solutions.

References

Anscombe, F. J., 1973, "Graphs in Statistical Analysis". *American Statistician,* 27 (1): 17–21.

Aggarwal, C, C., 2017, "Outlier Analysis", Second Edition, Springer, Cham., pp. 70-74

Ashkenas, J., 2012, "Underscore.js 1.3.1", Retrieved from *http://documentcloud.github.com/underscore*

Bostock, M., 2018, "D3.js Version 5.4.0", Retrieved from *https://d3js.org/*

Bostock, M., 2019, "Parallel Coordinates", *Mike Bostock's Block*, Retrieved from *https://bl.ocks.org/mbostock/1341021*

BWG, 2022, "Bankwesengesetz (Austrian Banking Act)", Version from 1.2.2022

Davis, J., 2021, "Parallel Coordinates", *Jason Davies's Block*, Retrieved from *https://bl.ocks.org/jasondavies/1341281*

Chang, K., 2012, "Underscore.math.js 0.1.2", Retrieved from *http://github.com/syntagmatic/underscore.math*

Chang, K., 2016, "Parallel Coordinates (0.7.0)", Retrieved from *https://syntagmatic.github.io/parallel-coordinates/*

Chang, K., 2021, "Nutrient Parallel Coordinates", *Kai's Block*, Retrieved from *http://bl.ocks.org/syntagmatic/3150059*

EZB VO (EU), 2016, "Regulation (EU) 2016/867 of the European Central Bank of 18 May 2016 on the collection of granular credit and credit risk data (ECB/2016/13)", *Official Journal of the European Union*, 1.6.2016

Galavotti, D., 2017, "Zoomable Circle Packing", *Davo Galavotti's Block*, Retrieved from *https://bl.ocks.org/davo/cd7261bd67581f284b6601fa4dd652b1*

GKE-V 2018, 2018, „Verordnung der Finanzmarktaufsichtsbehörde (FMA) über die Meldungen zur Erhebung granularer Kreditdaten (Granulare Kreditdatenerhebungs-Verordnung 2018 – GKE-V 2018)", *BGBl. II Nr. 170/2018*

Hartigan, J.A. and Kleiner, B., 1984, „A Mosaic of Television Ratings", *The American Statistician*, 38, 32-35

Holtz, Y., 2018, "Violin plot with jitter in d3.js", Retrieved from *https://www.d3-graph-gallery.com/graph/violin_jitter.html*

jQueryTeam, 2011, "jQuery v1.7", Retrieved from *http://jquery.com*

jQueryTeam, 2016, "jQuery UI - v1.12.1", Retrieved from *http://jqueryui.com*

Kosara, R., F. Bendix, and H. Hauser, 2006, "Parallel Sets: Interactive Exploration and Visual Analysis of Categorical Data.", *Transactions on Visualization and Computer Graphics (TVCG)*, vol. 12, no. 4, pp. 558–568

Leibman, M., 2012, "SlickGrid v2.1", Retrieved from *http://github.com/mleibman/slickgrid*

Li, J., J. Martens, and J. Wijk, 2010, "Judging Correlation from Scatterplots and Parallel Coordinate Plots", *Information Visualization* 9(1):13-30

Palmer, J., 2013, "d3.tip v0.6.3", Retrieved from *https://github.com/caged/d3-tip*

Petersson, J., 2018, "D3 Zoomable Scatterplot", Jonas Petersson*'s Block*, Retrieved from *http://bl.ocks.org/peterssonjonas/4a0e7cb8d23231243e0e*

Xing, Y., 2019, "parcoords-es v2.2.10", Retrieved from *https://github.com/BigFatDog/parcoords-es*

Interactive Visualization Tool

Outlier detection in large multidimensional datasets

Philipp Reisinger, Thomas Kemetmüller, Christoph Leitner

IFC Workshop (virtual event), 15 February 2022

Philipp.Reisinger@oenb.at
Thomas.Kemetmueller@oenb.at
Christoph.Leitner@oenb.at

www.oenb.at

Background

Oesterreichische Nationalbank collects monthly data from several hundred domestic banks for a **granular credit register (GCR)**.

GCR contains over **1 million credit instruments** with over a **hundred** categorical and numerical **dimensions** (ER model) per reporting date.

Requirements for **effective quality checks** increase with size and complexity of the collected data.

How can **incorrect data** be **identified**?

Quality Assurance

1. **Validation <u>rules</u>** (rejection of erroneous data on submission)

   e.g. volume ≥ 0, …

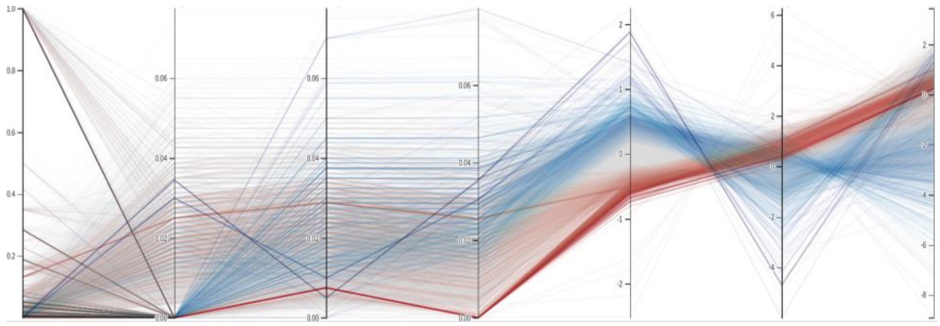
2. **Outlier detection using <u>statistical</u> models** (downstream)

   e.g. statistical inference, regression models, ...

3. **Outlier detection using <u>visual</u> methods** (downstream)

   e.g. parallel coordinates plot, scatter plot, box plot, …

Visualization



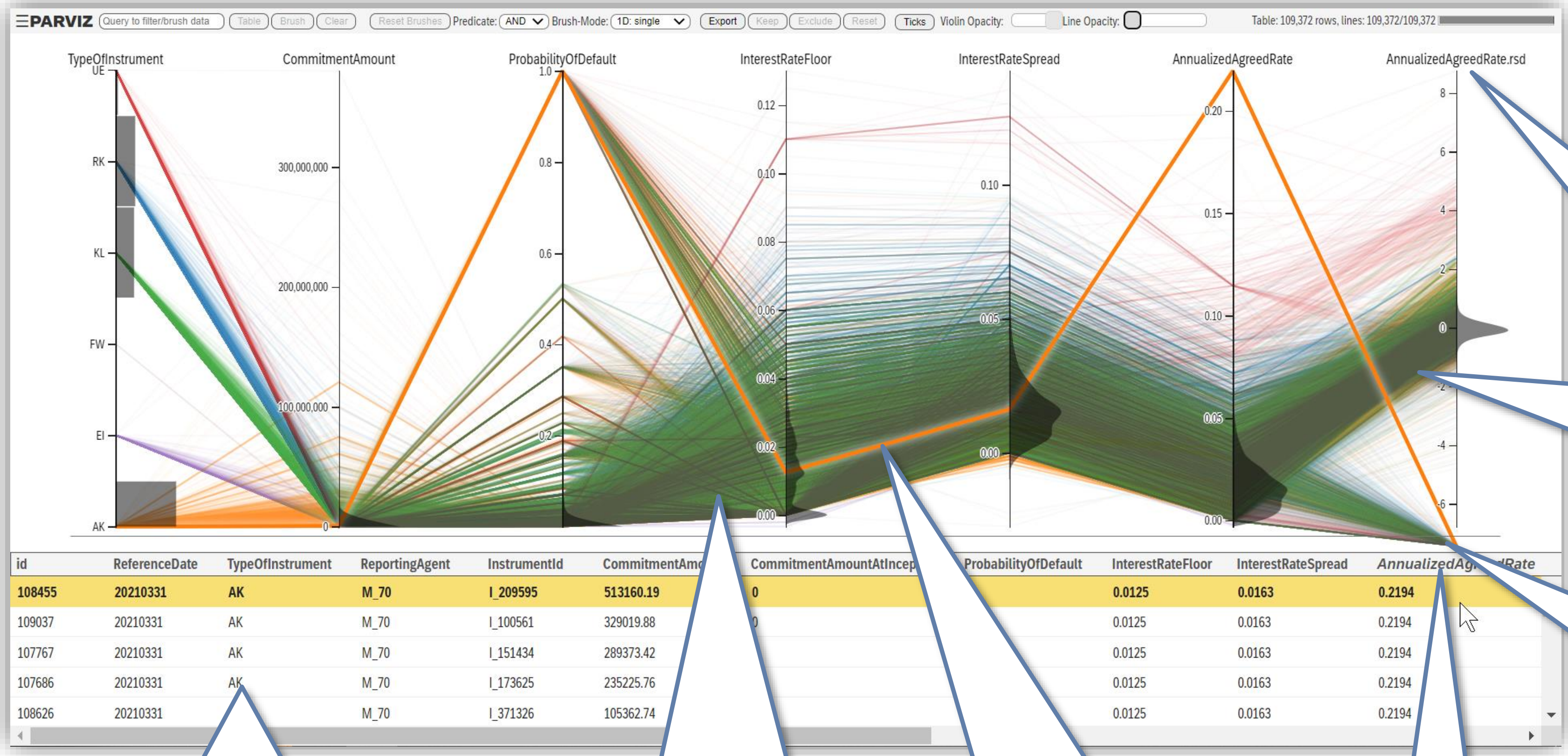
In **parallel coordinates plots**, each **observation** (credit instrument) is represented by a **polyline**, cutting the **axes** at its value in the corresponding **dimension** (colour can be used to highlight values of a certain dimension).

Parallel coordinates plots **require interactive implementations** to allow users to change the order and scaling of axes, polyline colour, ...

→ Customized tool for visual outlier detection:

- Pure front-end **web application** using standard web technologies (HTML, JS, CSS, …) plus additional libraries (D3.js, d3.parcoord.js, SlickGrid, …).
- Build upon **chart examples** from Mike Bostock, Jason Davis, Kai Chang, Jonas Petersson, Davo Galavotti.
- Set up as a **stand alone HTML file** (all libraries included) loading data from a **secured network folder**.

Visualization Tool: User Interface



Axes reordering (drag-and-drop)

Adjustable **polyline colouring**

Null values displayed below axes

Data **table** with **all dimensions**

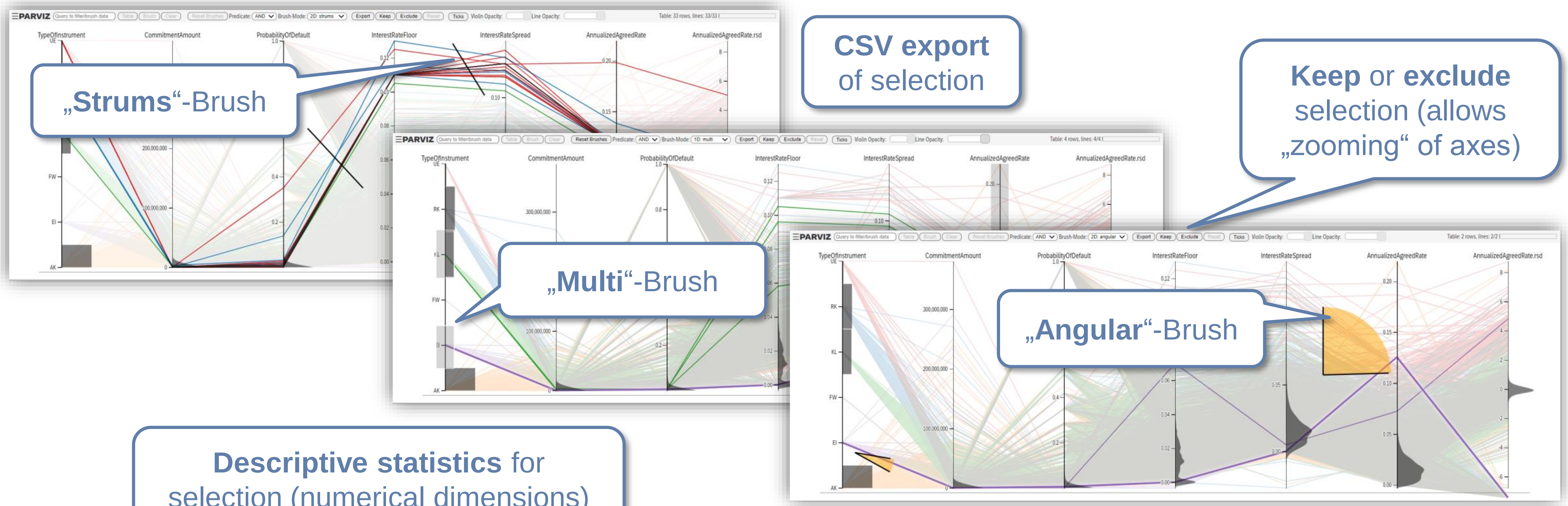
Parallel coordinates **plot** of **selected dimensions**

Polyline highlighting for data table rows

Sortable columns

Visualization Tool: Data Queries

Polylines can be selected („**brushed**“) in the **plot** with different **brush-modes** or via an **input box** for **precise** queries.



„**Strums**“-Brush

CSV export of selection

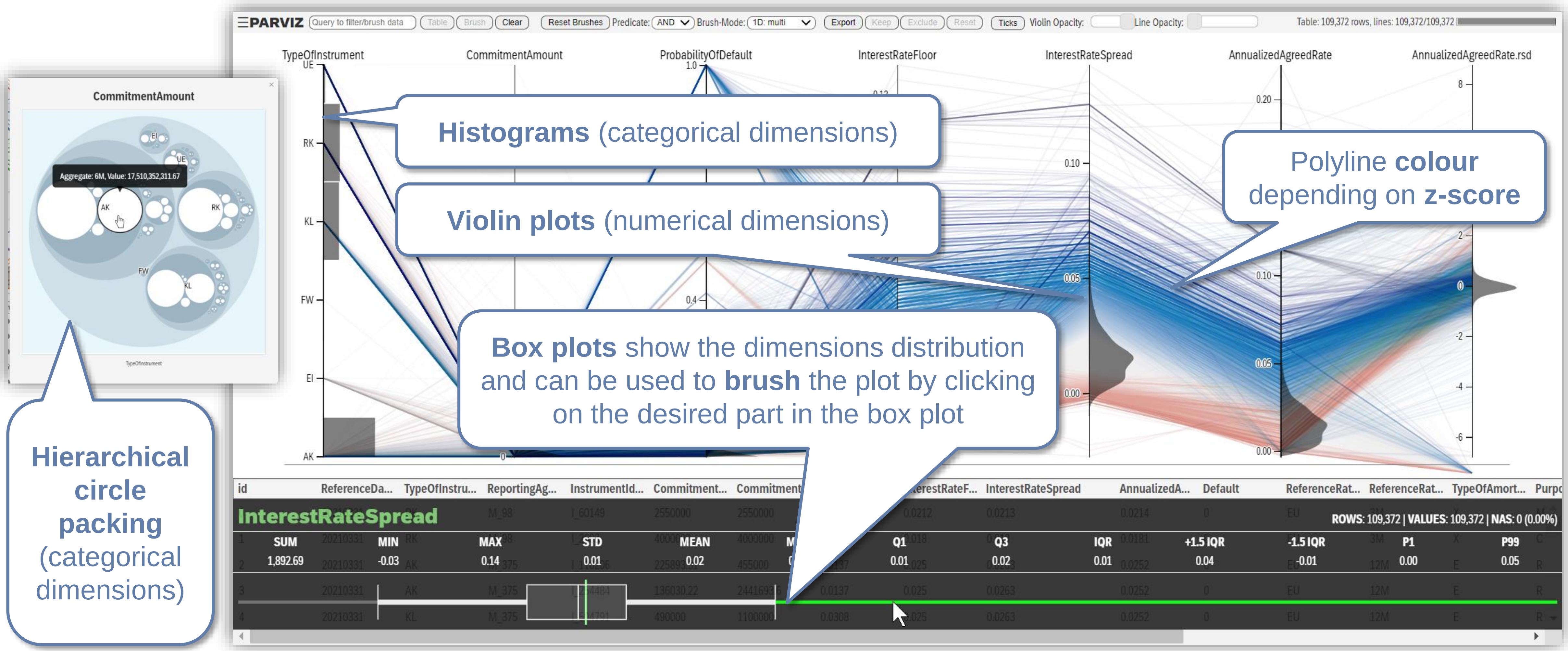
Keep or **exclude** selection (allows „zooming“ of axes)

„**Multi**“-Brush

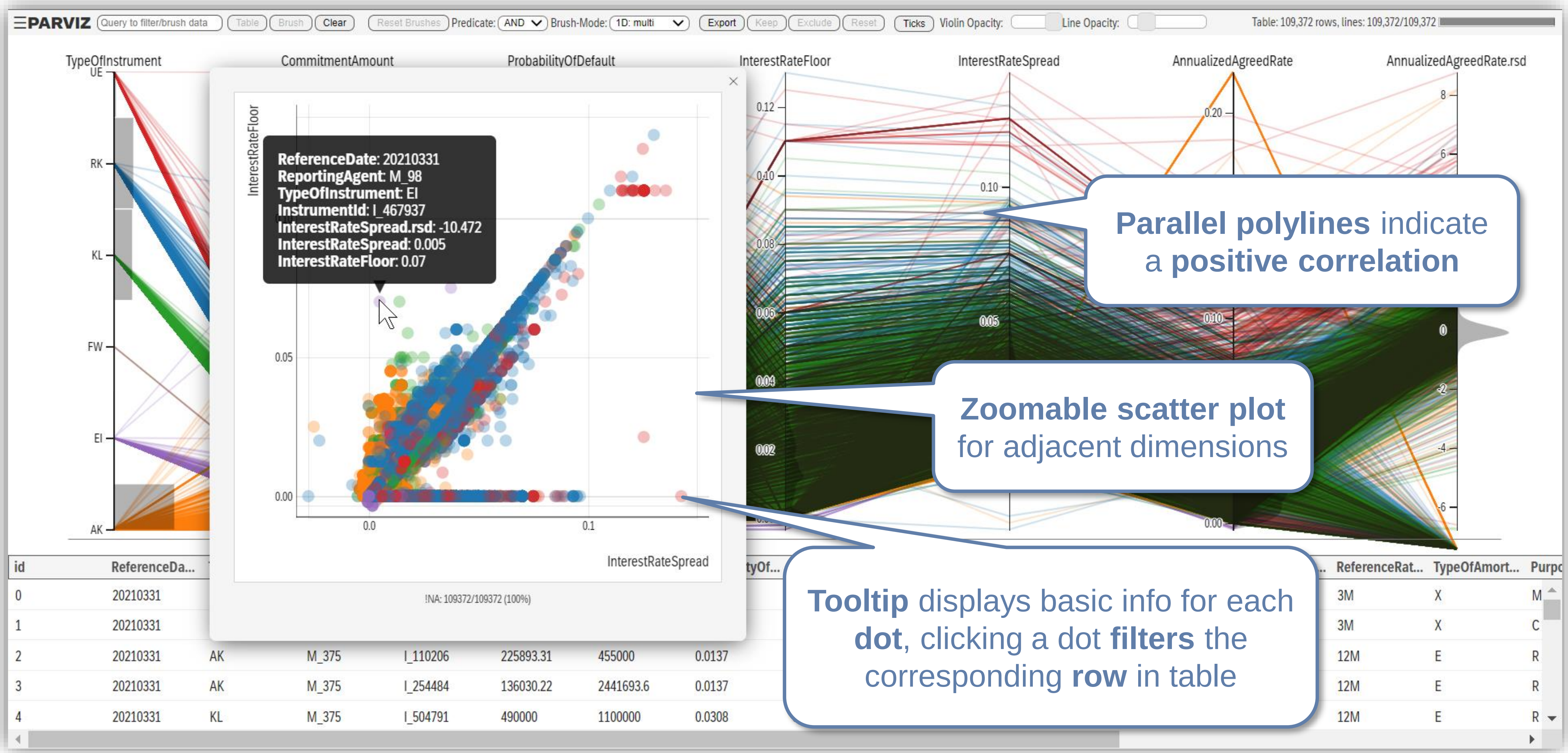
„**Angular**“-Brush

Descriptive statistics for selection (numerical dimensions)

Visualization Tool: Distributions

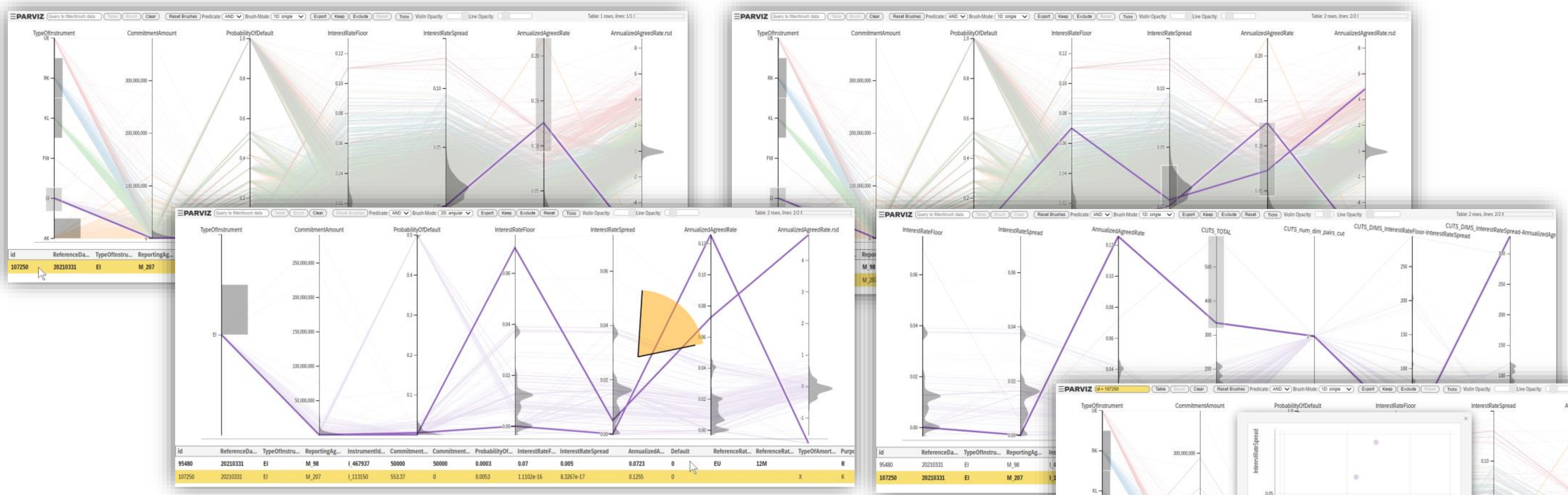


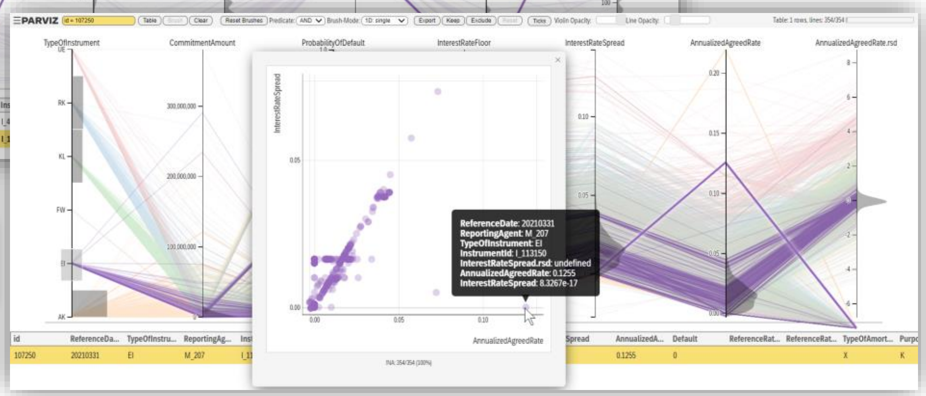
Visualization Tool: Correlations



Visualization Tool: Outlier Detection (example)

Features can be used independently or combined to find potential outliers…





… several ways can lead to the same results (suspicious data points).

Conclusion

- ✓ Interactive **parallel coordinate plots** are suitable to visualize **highly dimensional granular datasets** like the GCR.
- ✓ Various interactive visualizations for correlations and distributions help to quickly **grasp large datasets**, enabling users to **effectively identify potential data quality issues**.
- ✓ Visualizations improve the **understanding of complex statistical outlier detection methods**.
- ✓ Readily available and well documented **web development techniques** and chart examples **facilitate the development of tailor-made solutions.**

Danke für Ihre Aufmerksamkeit

Thank you for your attention

www.oenb.at

oenb.info@oenb.at

@oenb

@nationalbank_oesterreich

OeNB

Oesterreichische Nationalbank