PCP-Ed: Parallel Coordinate Plots for Ensemble Data

Elif E. Firat[a], Ben Swallow[b], Robert S. Laramee[a]

[a] *University of Nottingham, UK,*
[b] *University of Glasgow, UK,*

Abstract

The Parallel Coordinate Plot (PCP) is a complex visual design commonly used for the analysis of high-dimensional data. Increasing data size and complexity may make it challenging to decipher and uncover trends and outliers in a confined space. A dense PCP image resulting from overlapping edges may cause patterns to be covered. We develop techniques aimed at exploring the relationship between data dimensions to uncover trends in dense PCPs. We introduce correlation glyphs in the PCP view to reveal the strength of the correlation between adjacent axis pairs as well as an interactive glyph lens to uncover links between data dimensions by investigating dense areas of edge intersections. We also present a subtraction operator to identify differences between two similar multivariate data sets and relationship-guided dimensionality reduction by collapsing axis pairs. We finally present a case study of our techniques applied to ensemble data and provide feedback from a domain expert in epidemiology.

1. Introduction and Motivation

The Parallel Coordinate Plot (PCP), introduced by Inselberg [1], is a visual design showing multidimensional relations using parallel axes. PCPs facilitate data exploration and understanding relationships for multivariate data. One of the well-known challenges with PCPs is associated with overplotting. Rendering thousands of polylines causes overlapping edges that may obscure the underlying patterns in the image, especially in high data density areas [2]. We call a PCP with high-density areas resulting from many overlapping polylines a "dense" PCP. Ellis and Dix refer to this as, "too much

Email address: elif.firat@nottingham.ac.uk (Elif E. Firat)

data on too small an area of the display." [3] In these cases, interaction can be crucial in exploring the data and minimizing ambiguity. However, processing and analyzing overplotted data requires new approaches to support understanding. In our previous study on PCP literacy [4], we discovered that correlation between axes is one of the significant barriers to PCP understanding. This is one of the main inspirations behind the current work–to make the relationship between data dimensions clearer and more explicit. We believe the same concept could be applied to scatterplot matrices.

We propose novel visual feature and interaction methods to address challenges in PCPs that occur as a result of overlapping line segments. We introduce interactive glyph lenses that enable users to explore an overplotted area using a dynamic lens that hovers over the PCP based on mouse location. This interaction summarizes edges that intersect with the lens represented by arrow glyphs showing the average slope of a dense collection of edges. To convey relationships between dimensions, we display arrow glyphs placed below each adjacent pair of axes that indicate the correlation. We introduce a dimension reduction technique that enables users to evaluate a PCP by looking at the correlation value between neighboring axes and collapsing axis pairs that do not add information to the display. We also present a user option we call a subtraction operator, Δ, that displays the difference between two multi-dimensional data sets for quick comparison. The Δ operator addresses the unsolved problem of visually comparing multivariate ensemble data. In this paper, we specifically concentrate on interaction techniques for dense PCPs. The main contributions of this study are as follows:

- The introduction of interactive correlation glyphs for adjacent axis pairs
- Novel dynamic glyph lenses to support data analysis and comprehension
- A subtraction operator, Δ, to indicate differences between two multi-dimensional data sets
- Relationship-guided dimensionality reduction based on collapsing of axis pairs to reduce redundancy

We evaluate our methods with a case study based on the simulation of Covid-19 contagion behavior together with a modelling expert in this area. Visual comparison of ensemble data is considered an unsolved problem [5].

The rest of the paper is organized as follows: In Section 2, we review the previous work on reducing the impact of clutter in PCPs. In Section 3, we demonstrate interaction design including correlation glyphs, dynamic and static lenses, and the Δ operator. In Section 4, we discuss the performance of our visualizations and provide feedback from domain experts. Section 5 wraps up with conclusions and future work.

2. Related Work

Displaying a large multivariate data set in a 2D space has always been a challenge for data exploration due to over-plotting and clutter. We start by reviewing related surveys and focus on literature for the discovery of the information in dense and cluttered areas in PCPs.

Surveys: Dasgupta *et al.* [6] investigate different types of ambiguity in the PCP images and introduce a taxonomy for classifying them to reduce uncertainty. By creating a taxonomy, they aim to detect distinct sources of uncertainty in the design and link them to different impacts of uncertainty for the user. Similarly, Heinrich and Weiskopf [7] propose a taxonomy and assessment of strategies for modeling, visualizing, analyzing, and interacting with PCPs, as well as a classification of common tasks for investigation. Johansson and Forsell [8] summarize and categorize studies on evaluating PCPs. A thorough examination of previous research presents user-centered evaluations to report on the human-centered aspects of PCPs.

In this section, we focus primarily on previous work on PCPs that address visual clutter and ambiguity. We briefly introduce solutions to analyze large data on PCPs. In general, the methods for reducing the impact of clutter on dense displays can be categorized as frequency-based, using interaction and brushing, clustering, and edge-processing.

Frequency-based: Artero *et al.* [9] present a method for creating frequency and density plots from PCPs. The new plots enable interactive data exploration of large and high-dimensional data, enabling users to remove noise and highlight data-rich areas. Work by Geng *et al.* [2] proposes angular histograms and attribute curves that enable users to investigate clustering and linear correlations in large data sets to address over-plotting and clutter in PCPs. The state-an-art reported by Henrick and Weiskopf [7] has a particular subsection on frequency-based techniques that address aggregating edges together as an approach to overplotting and provides numerous methods for aggregating the data [10], [11], [12], [13] [14]. Our work incorporates a

frequency-based approach that counts the number of edge intersections with an interactive lens.

Interaction and Brushing: Blass *et al.* [15] present quantization and compression techniques for data pre-processing, as well as joint density distributions for adjacent variables enabling efficient GPU-based rendering of PCPs. In addition, they propose faster brushing methods for interactive data selection in several linked views. Raidou *et al.* [16] introduce a novel technique, Orientation-enhanced PCPs, to improve the view by visually enhancing segments of each PCP line emphasizing slope when there are several overlapping edges or when outliers and structures are obscured by noise. A novel effective selection method, the Orientation-enhanced Brushing (O-Brushing) is also presented that eliminates unnecessary user interaction. Another brushing method to enhance dense PCPs by Roberts *et al.* [17] introduces higher-order, smart data-driven brushing, and sketch-based brushing. The sketch-based brush is generated by connecting mouse clicks across the PCP on each axis at the chosen brush-axis intersection. Smart brushing assists the user during interaction by revealing patterns at run time. Some of our methods are based on interaction, however, none involve traditional brushing on PCPs.

Clustering: Data clustering is one method for reducing clutter in a PCP. Fua *et al.* [12] use hierarchical clustering to create a multiresolution representation of the data, and a variation on the PCP to express aggregated information for the clusters that facilitates navigation and filtering to explore the patterns and trends in the data. Ellis and Dix [18] propose several approaches for measuring occlusion by interactively adjusting the level of sampling. They explore three algorithms (raster, line, random) to measure the degree of occlusion. When compared to other algorithms, the raster algorithms result in higher accuracy. In addition to hierarchical clustering and calculating polyline occlusion techniques, Johansson *et al.* [19] use transfer functions to display different characteristics of clusters and transform each K-means-derived cluster to high-precision structural texture that, applied to a colored polygon, creates the cluster's final visual appearance. Blumenschein *et al.* [20] propose 30 different ordering strategies. The study introduces classification of task and pattern and investigates which PCP reordering strategies aid in detecting them. Our methods do not use explicit clustering. However, the lens we introduce summarize the edges that pass through them depicting average slope.

Edge-Processing: McDonnell and Mueller [21] introduce a technique

that shows each data point as a poly curve to facilitate edge bundling and declutter the display. Palmas *et al.* [22] present an edge-bundling technique that applies density-based clustering for each dimension. It represents the clustered lines as polygons, which reduces rendering time. They also use this strategy to enhance multidimensional clustering by developing attribute connections. Divino *et al.* [23] describe an edge bundling strategy used in PCPs to expose cluster information directly from the overview. The edge-bundling survey by Lhuillier *et al.* [24] presents a data-based taxonomy for classifying bundling methods and introduces a framework to describe the steps of bundling algorithms. Pomerenke *et al.* [25] render each line segment based on its slope between two axes in order to reduce the effect of cluttered lines. Horizontal lines are rendered with the default line thickness while diagonal lines are rendered thinner. The survey provides a subsection on PCPs and describes edge bundling papers that apply edge bundling for reducing the clutter and increasing readability [11], [21], [26], [27]. Our dynamic lens could be considered as a kind of edge processing technique.

In contrast to previous work, the techniques we describe generally focus on the space between axis pairs rather than on axes themselves. Most previous literature focuses on either the parallel axes or the polyline edges. We focus on supporting cognition of relationships between axis pairs in the context of dense PCPs. We introduce novel techniques to facilitate data analysis guided by correlation glyphs between neighboring axis pairs, showing the differences between data sets using a subtraction operator, and enabling the user to reduce dense areas and dimensions by collapsing axis pairs.

3. Fundamentals

In order to convey the strength of the correlation between axis pairs, correlation glyphs for each adjacent pair (Section 3.2) are presented in the PCP view. This provides users with a summary perspective of the multivariate relationships and an improved understanding of the link between axis pairs, which may not be visible by glancing at a dense set of edges. One of our techniques for dense displays is based on detecting the intersection of the edges with a glyph lens. The lens offers interactive feedback to the user as a function of the current mouse position that specifies center of the lens in the PCP (Section 3.3) in dense areas where the relationship between the axes may be difficult to interpret. The Δ operator (Section 3.4) is one of the techniques developed in order to understand the difference between two

comparable data sets. Also, axis pairs can be collapsed (Section 3.5) through a selection that enables users to view a reduced set of axes, motivated by redundant information.

Figure 1 shows an overview of the PCP tool we developed that allows a user to view different data sets via the user interface on the right of the screen (A). To demonstrate the relationship between each adjacent axis pair, correlation arrow glyphs are positioned under the PCP view (B). The figure also shows an example of a dynamic edge glyph lens (C) and some collapsed axes with stacked labels (D). The color scale on the left (E) is initially mapped to the edges on the first axis. This can be updated by selecting another axis. One of the user options offered by the tool is to display data labels and points where an edge crosses the axes by hovering the mouse over the edges and highlighting them. In addition, features such as rendering the average edge by taking the average of all edges and showing the zero point on the axes are also supported. See the demo video for complete details [28].

3.1. Ensemble Data from a Covid-19 Simulation

The ensemble data we study is a major motivation for the techniques we develop here. RAMP VIS [29] is a VIS volunteer group that responded to a call by the Scottish COVID-19 Response Consortium (SCRC) to support modeling scientists and epidemiologists [30]. The primary objective is to build a stronger and improved understanding of possible strategies to deal with the Covid-19 outbreak in the United Kingdom. We study the ensemble data set provided by the modelers by processing the large amount of simulation data given to the RAMP VIS group in our study. The data includes hundreds of time series for different regions of Scotland and different indicators (e.g., test, case, hospitalized, and fatality) and different age groups. The ensemble data is aggregated based on eight age groups and contains 23 parameters (see Figure 1). Each age group exemplifies an age interval (e.g. *Group 1* –> [age≤20], *Group 2* –> [20-29], ...,*Group 7* –> [70≤age], and *Group 8* –> Healthcare Workers) (See Appendix). The data contains the total numbers of susceptible, exposed, asymptomatic, symptomatic, hospitalized, recovered, deceased patients with a minimum, maximum, and mean values. Each age group is recorded on daily basis for 198 days. Each row in the data set represents a record of one day. See the Appendix for a more detailed description of the ensemble data.

By investigating the ensemble data in our novel PCP software, we aim to assist users in exploring models such that users can interactively compare

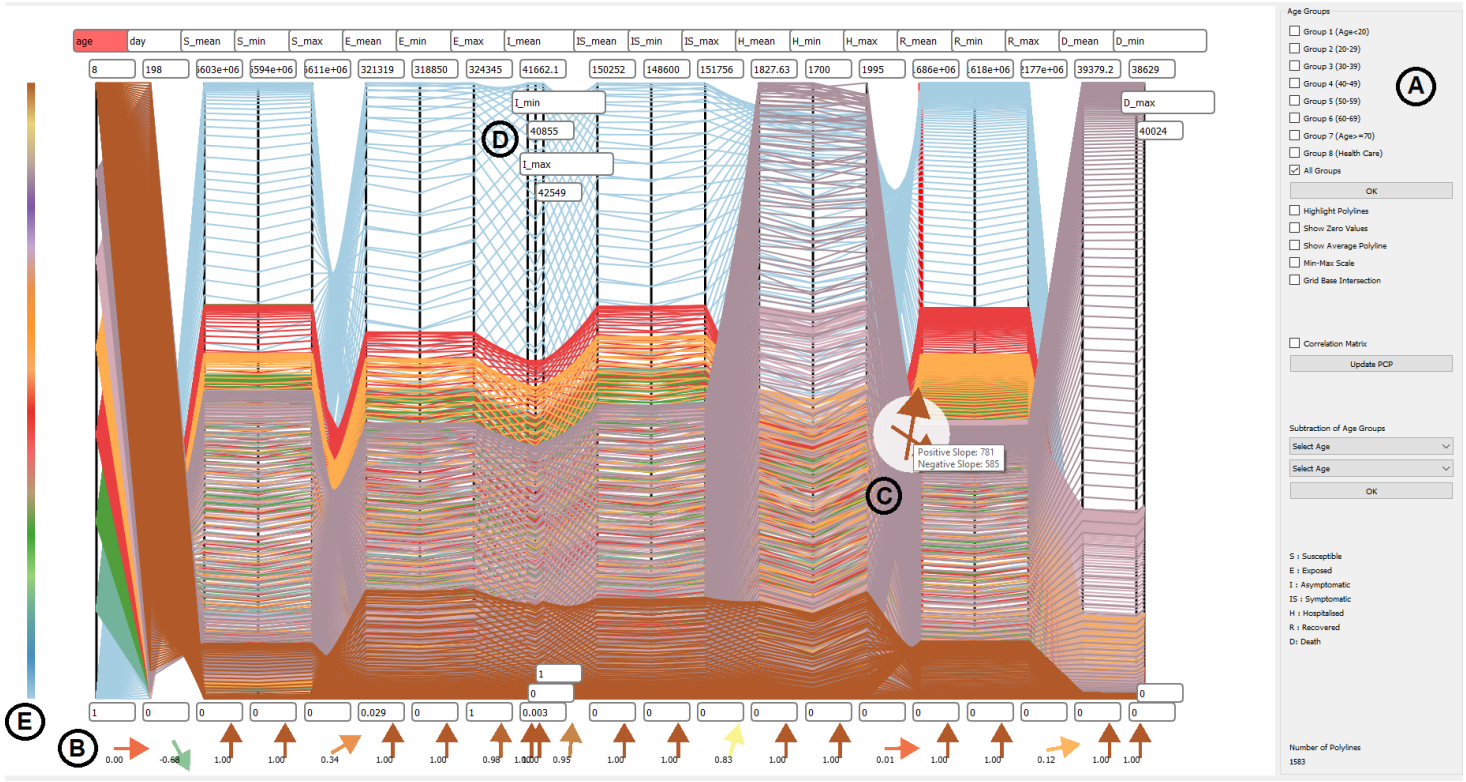

Figure 1: Overview of the PCP software tool. (A) The image displays user options, (B) the data with correlation glyphs under each axis pair, (C) interactive feedback in a dense area with an arrow glyphs lens, (D) collapsed axes pairs with stacked labels, and (E) a color legend. The PCP displays the predictions that the number of recovery in those under the age of 20 (Group 1) and the number of deaths in patients over the age of 70 (Group 7) will be higher than in other age groups. It also shows that mortality is lower for health care workers. The data contains of 1593 lines of records.

outcomes across age groups, identify differences between simulation parameters, and observe patterns as well as reveal outliers and features in the data.

3.2. Axis Correlation Glyph

The correlation coefficient is beneficial to identify relationships between the two variates. For some PCP examples, overlapped edges may create clutter and users may have difficulty viewing patterns between axes. Results of a previous user-study on PCP understanding reveal that identifying correlation can be a barrier to PCP literacy [4]. Deriving the slope of the edges and interpreting the links between data variables by looking at the PCP image can be challenging. Therefore, we introduce arrow glyphs for each pair of axes to present correlation values explicitly (see Figure 1, (B)). This offers users a convenient way to interpret the relationship between two dimensions by glancing at the correlation glyphs. Many-to-many PCP is an alternative design to show the axes correlation, for example, the many-to-many design of Wu *et al.* [31] or Lind *et al.* [32]. We believe that our glyph-based technique could benefit these visual layouts as well. However, many-to-many axis layout is difficult to scale as evidenced by the low number of dimensions.

The appropriate design of glyphs is critical for usability and successful visual communication. Relevant visual channels should be carefully selected and integrated for an effective glyph design [33]. The study by Fuchs *et al.* [34] methodically gathers and categorizes the literature on data glyphs, describing their designs, questions, data, and tasks. The arrow glyph is included in the "One-to-One Mapping" category. Borgo *et al.* [35] describe that glyph design can use a variety of visual channels, including shape, color, texture, size, and orientation. Our glyph design reveals the relationship between axes-pairs by presenting an arrow shape, using a peer-reviewed color library [36] and direction of the slope for the correlation value. In addition, the color was consistent with the polylines and the color scheme used in the PCP has also been adapted to the correlation glyphs based on κ.

Design Justification: For dense PCPs, it may be difficult to determine relationships between data dimensions by observing the slope of the edges. We use an arrow glyph that conveys correlation value using slope information. The arrow glyph reveals the trend between dimensions using both the slope and direction. There are several other options possible here. Both bar charts and pie charts can encode the same information such as a number of intersecting edges and average slope. However, we wanted to map slope of edges to a glyph with slope intuitively built in. Arrow glyphs already

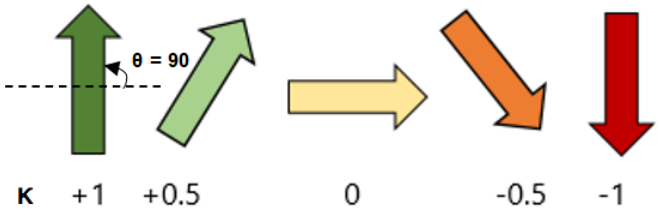

Figure 2: The figure shows the glyphs that represent the correlation coefficient value between adjacent axis pairs displayed in the $\theta \in$ [-90, +90] range. The color scale can be modified by the user.

have these characteristics naturally build in whereas other charts and glyphs generally do not.

The correlation values, κ, are calculated using Pearson's Correlation Coefficient [37] for each axis pair. The arrow glyph represents each pairwise coefficient value. The individual distributions of the two related axis pairs are shown in the range $\kappa \in$ [-1, +1] and the arrow glyphs represent the range $\theta \in$ [-90, +90] and correspond to the correlation values, κ, indicating negative and positive relationships respectively (see Figure 2). In addition, the color scheme used in the PCP has also been adapted to the correlation glyphs based on κ.

3.3. Dynamic Edge Glyph Lens

The underlying structure in the data is not always obvious in PCPs. The dense PCP resulting from overlapping of the edges may cause information to be covered. This may make it difficult for the user to interpret the existing correlation and observe patterns. Thus, we introduce a glyph lens designed to reveal information that may be obscured by edge overplotting. Observing the dynamic glyph by hovering the lens over the edges offers the user a summary of the edges and of the average slope, θ_{AVG}, of the edges represented by arrows.

Design Justification: This is a special type of lens that focuses on the space between the axes as opposed to the axes themselves. Frequency-based approaches previously presented in the related work focus primarily on axes instead of relationships between axes. Our dynamic edge glyph lens solution offers a user interaction-based feature integrated into the PCP to uncover the trends between axes and improve the interpretation of the data (see Figure 3). We chose the same arrow glyphs as in Figure 2 because they intuitively

encode slope and thus the correlation between axes. Other charts and glyphs can encode this same information but not intuitively because the slope is not the predominant characteristic of most charts and glyphs, e.g., pie charts, bar charts, etc.

To address the overlap problem, we focused on the intersection of the edges with the lens, starting from the left axis and ending on the right axis (in any pair). The dynamic edge glyph shows the number of edges that intersect with the lens and average slope, θ_{AVG}, of each intersecting edge (see Figure 1, (C)). After calculating θ_{AVG}, the edges intersecting the lens are grouped according to whether the edge has a positive or negative slope. The two groups are represented by two arrows placed in the lens glyph (see Figure 3a). The upward arrow in the glyph lens represents the average positive, $\theta_{\text{AVG}}+$, and the other represents the average negatively sloped edges, $\theta_{\text{AVG}}-$. The resulting arrows are designed similar to the correlation glyph arrow (Section 3.2). They display the angle, $\theta \in$ [-90, +90] by calculating the average angles of inclination $\theta_{\text{AVG}}+$, $\theta_{\text{AVG}}-$ (see Figure 3b). The magnitude of the arrows is also scaled by the number edges (with positive and negative slope) that intersect with the lens. The color of the arrows is mapped to the color legend provided. This interactive feature facilitates uncovering hidden correlation information between data axes by hovering the lens and observing the trends in the data (see Figure 4).

3.4. Multivariate Subtraction Operator, Δ

Plotting two data sets on the same PCP or two adjacent PCPs is a common approach for comparison. However, both of these can lead to challenges with large data sets as both may be dense to start with. We introduce a multivariate subtraction operator, Δ, that we can apply to compare two similar data sets on the same PCP.

Design Justification: In our case, we have ensemble data from a Covid-19 simulation, thus, the simulation configurations are directly comparable. The Covid-19 simulation data is major inspiration for our features because the modelers are very interested in comparing different simulation configurations. The Δ operator reveals the differences between similar data sets e.g., the case of ensemble data. The variation between data attributes such as hospitalization or recovery numbers can be interpreted quickly. Plotting the difference S_Δ between two simulations, S_1 and S_2, in the same space as S_1 and S_2 themselves is simple, fast, and intuitive.

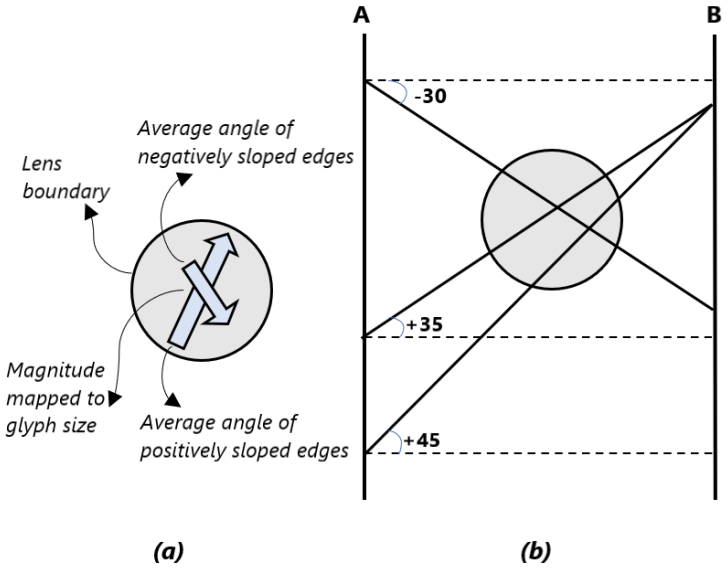

Figure 3: An overview of (a) the glyph lens, (b) edge intersection summary with the dynamic edge glyph lens. This figure shows two attributes in the PCP and three line edges that connect A and B. After the detection of the intersecting edges for both, arrows are shown as in the lens (a) representing the edges. Since there are two positively sloped and one negatively sloped edges showing the relationship between A and B, the arrow representing the positive slope is longer than the other as it indicates two edges.

In order to perform the multivariate subtraction, the attributes of the data sets are the same and in the same order, such as the Covid-19 simulation [30] we use. The edges of the difference obtained after the subtraction can also be rendered and shown in the PCP (see Figure 5). As a result of plotting the difference data, S_Δ, labels for minimum, d(min), and maximum, d(max), values are updated.

The subtraction operator, Δ, is implemented to highlight changes in simulation output parameters for different input configurations that may or may not be similar. We perform subtraction on two configurations selected through the user interface (see Figure 1 (A)). The second selected, S_2, is subtracted from the first, S_1. This operation is applied by subtracting the corresponding values in the same dimensions. Given a simulation, S, with dimensions $S(d_0, d_1, ..., d_n)$ the subtraction operator computes the difference, Δx, between data values, x, that correspond to one another e.g.,

$$S_\Delta = S_1(d_n(x_m)) - S_2(d_n(x_m))$$

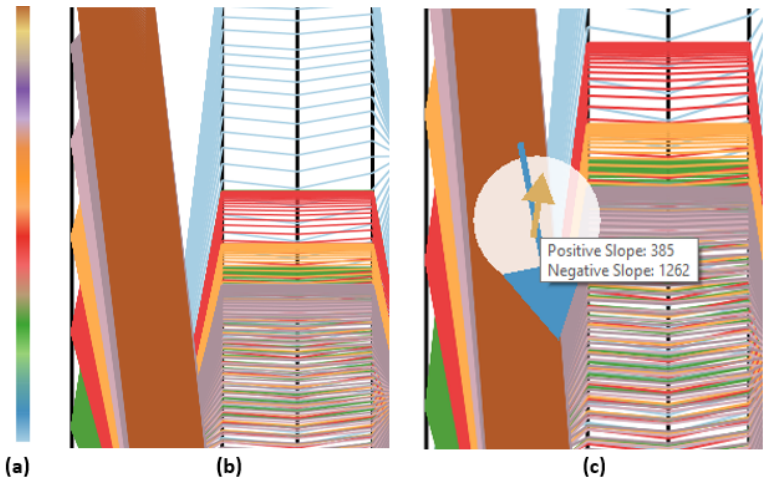

Figure 4: An overview of (a) a color legend, (b) a dense area in the PCP, and (c) summary of edges in the same area with dynamic edge glyph lens (see section 3.3). The numbers indicate the number of edge intersections with the lens.

Where d_n is a given dimension and m is a given data index. With the selection of S_1 and S_2, the maximum value of the axis, d(max), is derived as the maximum value, d(min), of both $S_1(d_n)$ and $S_2(d_n)$, and the minimum value is set as $-1 \times d(max)$. The S_Δ obtained as a result of subtraction is plotted on the PCP. Positive or negative differences can be seen within the updated d(min) and d(max).

Figure 5 displays the output of the subtraction operator, Δ, applied to *Group 1* ([$\leq$20]) and *Group 7* ([70$\leq$ d_{age}]) provided in the Covid-19 simulation [30]. The calculation is performed by subtracting *Group 7* from *Group 1* plotted with polylines in yellow and green respectively. We can see an example of this by looking at the age group dimension. By subtracting the values of d_{age}, the result is -6 *(1-7 = -6)*. The edges representing the difference between two data sets are plotted within age groups $\in$ [-8, +8], shown in red. Green points on each axis indicate zero values for each dimension and enable viewing the negative differences. The result is shown in Figure 5. The number of hospitalizations, h, and deaths, d, in patients over 70 years of age is much greater than in patients under 20 years of age.

3.5. Dimensionality Reduction by Collapsing Axis Pairs

The purpose of using parallel coordinates is to expose particular features in the multivariate data. However, the essential information sometimes may

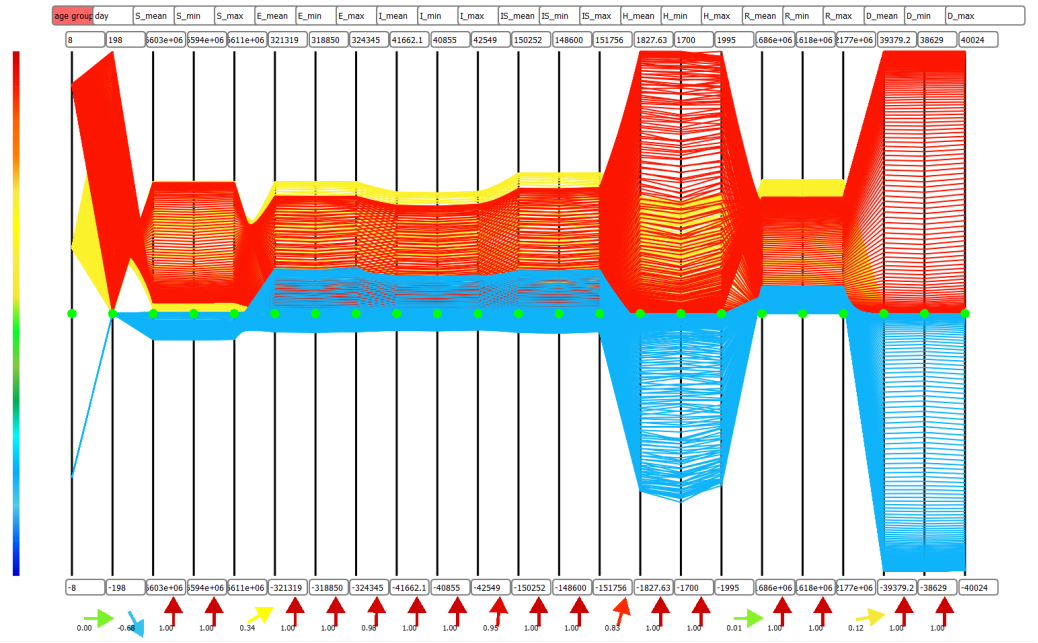

Figure 5: Multivariate subtraction performed on the *Group 1* ($[d_{\text{age}} \leq 20]$) and *Group 7* ($[70 \leq d_{\text{age}}]$) in yellow and red respectively. The difference, Δ, is shown in the PCP with blue polylines. Using Δ, multivariate differences between age groups become obvious with respect to hospitalizations, h, and mortality, d. Green points on each axis address zero values on the axis.

not be obvious due to overlapping edges and a high number of dimensions plotted in the PCP. The images vary depending on the order of axes. In order to display the relationship between dimensions, we use glyphs showing the tendency between each axis pair and the corresponding correlation, κ, (see Section 3.2). By using on these correlation glyphs, the user may exploit relationship-guided dimensionality reduction via collapsing of axis pairs.

Design Justification: The high-dimensional ensemble data is based on eight age groups and contains 23 parameters with minimum, maximum and mean values of each indicator. The data includes repetitive information. We introduce this user option that gives a different perspective on the data dimensions by removing some of the redundant elements that do not add new information to the PCP. The objective of collapsed axis pairs is to decrease the number of dimensions and depict a less complex PCP view e.g., especially for values of $\kappa = 1$. This feature enables the user to explore and display the relationship between dimensions, d, that they choose to emphasize and with

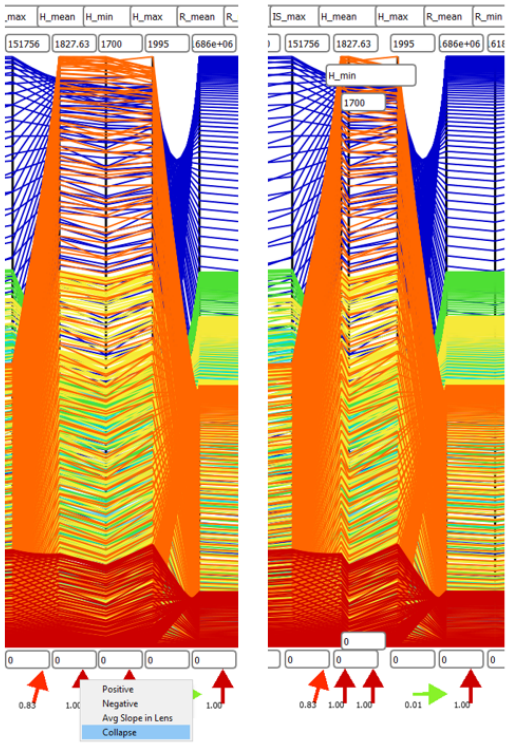

Figure 6: The collapsing of the h_{mean} and h_{min} axis pair by right-clicking on the correlation glyph showing the relationship between them. The labels of h_{min} are stacked to indicate the collapsing process.

less redundant information (see Figure 1 (D)).

The user option provides a new view of the data dimensions by reducing some of the redundant dimensions that do not present a particularly notable pattern in the PCP. Collapsing axes can be guided by observing correlation glyphs. For example, pairwise axes with a correlation κ of unity may be collapsed without loss of information. The process is performed by right-clicking on a correlation glyph for a given axis pair and reducing the space between them by translating the right axis closer to the left axis. In the new layout, the axis name and maximum value labels of the right axis are stacked under the left labels of the pair axis while the minimum label is placed on the top of other minimum value labels. The collapsing procedure can be undone by right-clicking on the same correlation glyph to obtain the previous PCP view.

Figure 6 demonstrates an example of axis pair collapsing between h_{mean}

and h_{min} (h: *Hospitalization*). Selected collapsed axis groups are data variables with, $\kappa = +1$, in other words, showing a direct relationship. As a result of the collapsing of an axis pair, two dimensions are positioned side-by-side and axis labels stacked on top of each other are displayed. Figure 1 (D) shows an example where 3 dimensions are juxtaposed after collapsing two-axis pairs. With the dimensionality reduction feature, redundant and repetitive information that makes it more challenging to reveal patterns in the data can be excluded.

Additional Features: In addition to the previous features we introduced, the software includes features that are helpful in exploring the ensemble simulation data. We provide a feature that allows the min and max labels to be updated such that the axis data in a given range can be scaled. We offer six different color scales for color mapping in the PCP using a color library by Roberts *et al.* [36]. We also introduce the features of drawing the average polyline using the average of the edges, or rendering the positive and negative sloped edges by right-clicking on any area of the PCP, using focus+context. Finally, we developed a κ matrix to understand the relationship between each data dimension combination. In the matrix, the user can select one of the dimensions and sort the correlation values from smallest to largest.

4. Evaluation

We provide three use cases to evaluate our techniques and provide a demo video for these use cases. We demonstrated the software to the domain expert and reported feedback collect from the expert in this section. See the demo video for details [28].

4.1. Case-Study

In this section, three use cases demonstrate the effectiveness of our techniques in understanding underlying trends in the Covid-19 ensemble data.

Use Case 1: Multivariate Comparison of Age Groups To explore the multivariate differences between age groups, we used the Δ operator between two age groups in the first simulation configuration presented in Figure 1. For example, we render the relationship between the simulation results under age 20 (Group 1) and above age 70 (Group 7) (see Figure 5) by applying the Δ operator to these age groups. We observe that the hospitalization and mortality numbers are much higher compared to Group 1.

Use Case 2: Comparing Input Parameter Values, p_{inf} Probability of infection, p_{inf}, is one of the most interesting input parameters of the simulation according to the simulation domain experts. We selected the two simulations with the minimum and maximum, p_{inf} (min) and p_{inf} (max), for input parameter values. Then we utilized the Δ operator to compare the outcomes for these two simulations to investigate how influential the p_{inf} parameter is and understand how input parameter values influence the output. To compare two simulations, we sorted simulations by the p_{inf} value and included all age groups in Simulation 3 with the lowest p_{inf} value and Simulation 101 with the highest p_{inf}. We then used Δ operator to render the difference between these simulations. As a result of Δ, Simulation 101 shows a very clear difference for all output parameters compared to Simulation 3 (see Figure 7). The Δ operator indicates that p_{inf} is a very influential input parameter.

Use Case 3: κ-guided Dimensionality Reduction We examine the PCP in Figure 1 and the correlation glyphs under each axis pair. We observe that there is always a direct relationship between the mean, min and max values of each parameter in the output. We used this observation to reduce the redundant dimensions and produce a new image with the redundant axes removed. The dimensionality reduction technique we utilize by collapsing axis pairs results in an image that reduces the number of dimensions by almost 50% in the PCP (see Figure 8). Note that the pairwise glyphs are also preserved and remind the user of the redundancy.

4.2. Domain Expert Feedback

This work is partially carried out in collaboration with Ramp Vis [29] team, who support the modelling scientists and epidemiologists in the Scottish COVID-19 Response Consortium (SCRC) [30] (see Subsection 3.1). We had three meeting sessions, including visualization experts, modellers, and statisticians. The brainstorming sessions facilitated understanding of the data simulations and exploring the most influential input parameters. We organized a feedback session and interviewed Dr Ben Swallow, with a PhD in Statistics and working in the School of Mathematics & Statistics, University of Glasgow. He has been working in statistical simulation and estimation for seven years and has spent approximately four years on epidemiological studies. Some of his work focuses on Bayesian parameter inference and model selection and methods for zero-inflated data. Our interview questions were adopted from Hogan *et al.* [38].

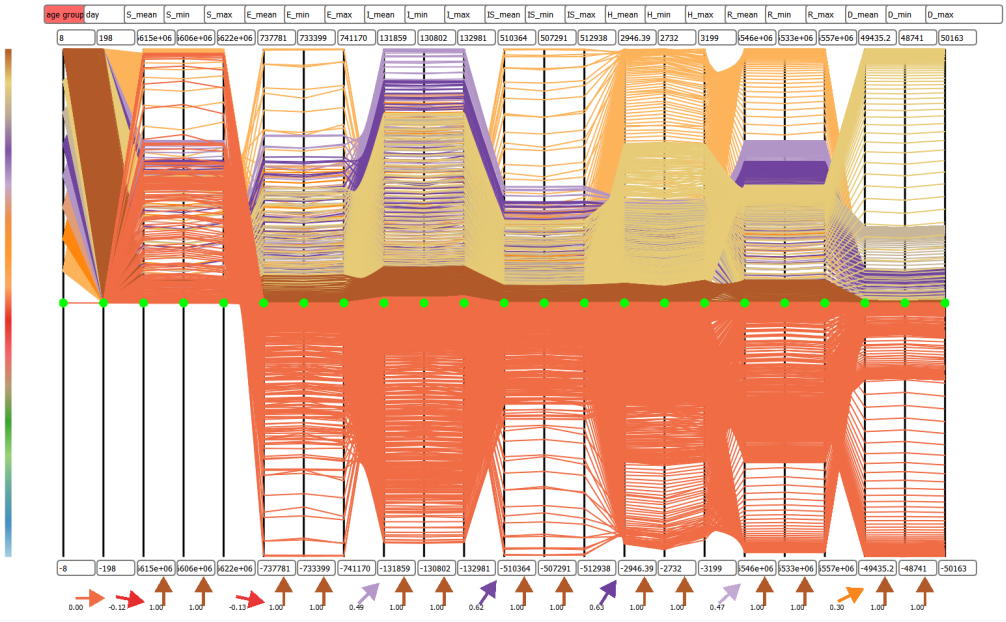

Figure 7: This figure displays the subtraction operator applied to Simulation 3 with lowest p_{inf} and Simulation 101 with highest p_{inf}. The color is mapped to the first axis [-8, +8] of the PCP.

Correlation Glyph: We demonstrated the correlation glyphs, and he reported: *"It's a really good way of guiding the dimension reduction when you have so much information. Users are trying to find a way of deciding how to reduce it down and extract information. It's pretty cool."*

Dynamic Edge Glyph Lenses: When we presented the both glyph lenses to watch the behavior of the glyphs and discover areas with a lot of variation, he stated that the feature is useful and added; *"I think it's just another way of looking at the kind of sensitivity to that particular parameter and in what direction it's going. I particularly know the type of people that would likely use this. I think you can get this through more hardcore mathematical sensitivity analysis, but I think getting an idea of a sensitivity across regions of parameters and different parameters will be very welcome. It would be huge benefit of having this type of software. Yes, I really like that."*

Dimensionality Reduction: We mentioned that there are a lot of redundant dimensions in the data and to the expert. He agreed on this and reported: *"Yes, that's what we found from the mathematical analysis as well. It was p_{inf} and P_s that we really the only two parameters that had any impact*

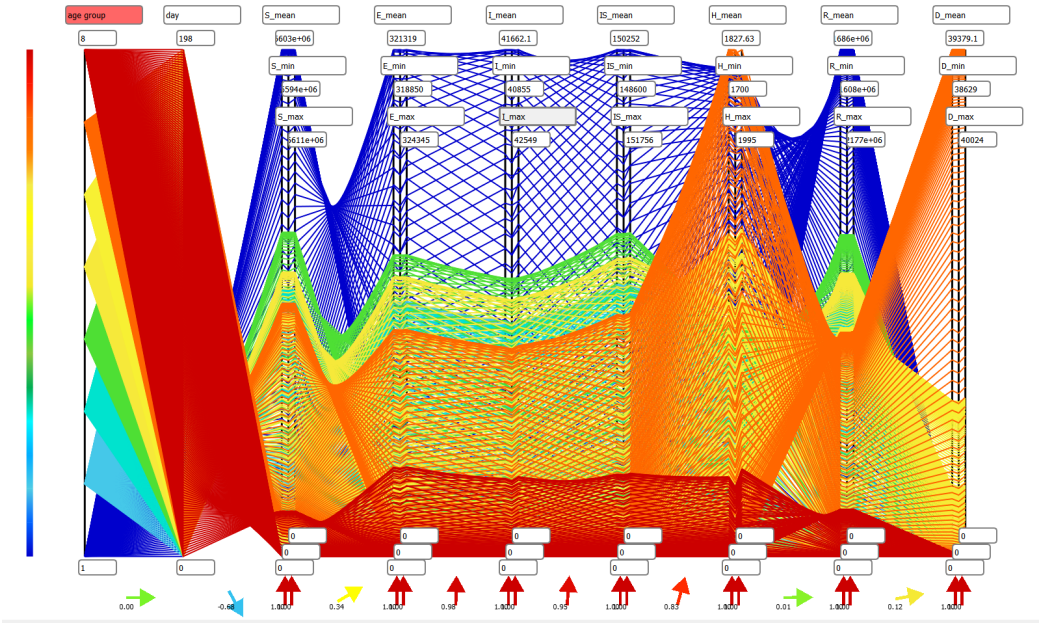

Figure 8: This figure demonstrates dimensionality reduction applied on axis pairs with $\kappa = 1$.

at all. It seems that that's what's being visualized here and in a much more clear manner."

Use Case 1: Comparing Age Groups When we first demonstrated the subtraction operator, Δ, to the expert, he liked the concept of presenting the difference between two multidimensional data sets visually to compare them and stated: *"I think it's highlighting differences. The differences are going to be specific to particular groups or compartments of the model. So I like being able to observe that. From a policy point of view, you think "if I change this parameter, what's it going to change?" If it has a negative impact on say younger people, in terms of the number of cases, but maybe it reduces the deaths in another age category, then that's going to be useful from a policy perspective rather than just saying, "well, we've just looked at the combined groups". There's going to be more cases in group two. You know group two is going to be less impacted by Covid-19 in general. And knowing how it's affecting things in a more detailed and visual way, I think, is really useful."*

Use Case 2: Comparing Input Parameter Values, $\mathbf{p_{inf}}$ The p_{inf} input parameter has a significant influence on the outcome, and the differ-

ence between simulations verifies that. We demonstrated this in Figure 7 and asked the expert if he finds this helpful. He commented: *"I would like confirming what we have done already [formal mathematical sensitivity analysis], or if we used the software first and looking at what we think might be the most important parameters. You know most of the model developers will have an idea of which is the most important parameters are. Visualizing that is very useful for confirmation."*

We also asked the expert how he figured out the most influential parameters without the software and how long it takes. He reported: *"We normally have to do a full mathematical sensitivity analysis of the model. You could look at things like histograms of the output, so they would tend to be either visualized viewpoints, but probably nowhere near as sophisticated as this. Or kind of formal mathematical way you look at things like the derivative, i.e., changes in the output as a function of the different parameters. But that's a lot more complex and time-consuming than this. The process really depends on the complexity of the model and number simulations you have to do, but it would take probably at least a couple of hours to run mathematical analysis. Because you generally use a Monte Carlo approach across lots of simulations as you are plotting here. But again, there are lots of different questions that you could ask using this the PCP software in terms of the sensitivity across time, different age groups, and different classes. You would have to do it on the separate simulation or sensitivity analysis for each of those different configurations, whereas here you have the option of interrogating them all in one go or very quickly switching between the different questions that you might want to ask of them to the model."*

Use Case 3: κ-guided Dimensionality Reduction Dimensionality reduction and axis ordering are still considered unsolved problems. We demonstrated our κ-guided dimensionality reduction features by collapsing axis pairs (see Figure 8). We asked the expert if the feature let him see anything that he might previously have not been able to see or make some new observations or hypotheses. He reported: *"One of the common aspects of these types of models is over parameterization. When you try and estimate the parameters, if the model is not sensitive at all to the input parameters, then no matter how much you try and make any inference, it is not going to be useful at all. So from that perspective, I think this feature is useful. The standard approach to deal with overparameterization is that if you have got parameter redundancies to make some model reduction - that's quite complex to do without a good understanding of the model and where it is lack of sen-*

sitivity arises. So I think it would be really helpful in deciding how to think about either combining outputs into a single one. For example, if there was age differentiation or there was no impact on the parameters on age, then I think you would see that here and in terms of looking at the different compartments, I think that is really useful. Parameter redundancy is generally quite a useful way of guiding model reduction and that would be very helpful there."

We asked the expert if the feature might increase confidence in terms of the correctness or accuracy of the simulations. He stated: *"Yes, I'm sure. If you are seeing some of the maximum numbers, if you knew, for example, that hospitalizations never got above a particular point but your model is consistently estimating numbers of hospitalizations to be in the hundreds of thousands, and you know that's not realistic, you would probably have some lack of confidence in that model. I think that could be something else that this helps with. In terms of focusing where you perhaps want to do data collection as well, if you know there's a lot of sensitivity. It seems like hospitalization in this model are a very sensitive, very valuable output. Then you might try and focus your data collection on that when you want to make inferences and try and estimate these parameters. That would probably be a good way of guiding that decision as well."*

5. Conclusion and Future Work

We present interactive glyph lenses, which enable users to explore an overplotted image with a dynamic lens that hovers over the PCP using mouse position. This interaction outlines the edges that overlap with the lens, represented by arrow glyphs that show the average slope, θ_{AVG}, of a dense collection of edges. We display an arrow glyph below each adjacent axis pair that indicates the correlation between dimensions. We present a dimension reduction technique that allows users to simplify a PCP based on the correlation value, κ, between adjacent axes and collapsing axis pairs that do not add information to the display. We also provide a user option we call a subtraction operator, Δ, which displays the difference between two multidimensional data sets for comparison. We evaluate our techniques with a case study based on a simulation of Covid-19 in collaboration with a modeling expert.

One limitation we encounter with a dynamic lens is run-time edge detection, which may slow down when there are too many edges. In addition, with

large data sets, the performance of detecting edge intersections starts to degrade. In the next step, pre-computing a summary of edge intersections in a static grid and then displaying the meta-data, rather than trying to calculate the edge intersections at run-time, is a way to manage this challenge. We also note that the subtraction operator is currently limited to (and targeted at) ensemble data. A more generalized version remains future work.

Future improvements addresses limitations e.g., sorting axis pairs based on correlation value in ascending order and updating the PCP view accordingly. However, plotting axis pairs according to pairwise, κ, order is not feasible with the traditional PCP axis ordering. Therefore, introducing a new axis plotting approach to convey the axes' relationships is a future endeavor. Another limitation is scalability, i.e., how to arrange axis labels when 10 or more pairs of neighboring axes are collapsed. Future directions address other limitations such as, including multiple lenses, adjustable lens size, and additional operators, such as addition, multiplication and division of simulation data sets.

References

[1] A. Inselberg, B. Dimsdale, Parallel coordinates: a tool for visualizing multi-dimensional geometry, in: Proceedings of the 1st IEEE Conference on Visualization, 1990, pp. 361–378. doi:10.1109/VISUAL.1990.146402.

[2] Z. Geng, Z. Peng, R. S. Laramee, J. C. Roberts, R. Walker, Angular histograms: Frequency-based visualizations for large, high dimensional data, IEEE Transactions on Visualization and Computer Graphics 17 (12) (2011) 2572–2580. doi:10.1109/TVCG.2011.166.

[3] G. Ellis, A. Dix, A taxonomy of clutter reduction for information visualisation, IEEE Transactions on Visualization and Computer Graphics 13 (6) (2007) 1216–1223. doi:10.1109/TVCG.2007.70535.

[4] E. E. Firat, A. Denisova, M. L. Wilson, R. S. Laramee, P-lite: A study of parallel coordinate plot literacy, Visual Informatics 6 (3) (2022) 81–99. doi:10.1016/j.visinf.2022.05.002.

[5] J. Wang, S. Hazarika, C. Li, H.-W. Shen, Visualization and visual analysis of ensemble data: A survey, IEEE Transactions on Visualization

and Computer Graphics 25 (9) (2019) 2853–2872. doi:10.1109/TVCG.2018.2853721.

[6] A. Dasgupta, M. Chen, R. Kosara, Conceptualizing visual uncertainty in parallel coordinates, The Eurographics Association & John Wiley & Sons, Ltd. 31 (3pt2) (2012) 1015–1024. doi:10.1111/j.1467-8659.2012.03094.x.

[7] J. Heinrich, D. Weiskopf, State of the Art of Parallel Coordinates, in: Eurographics 2013 - State of the Art Reports, The Eurographics Association, 2013. doi:10.2312/conf/EG2013/stars/095-116.

[8] J. Johansson, C. Forsell, Evaluation of parallel coordinates: Overview, categorization and guidelines for future research, IEEE Transactions on Visualization and Computer Graphics 22 (1) (2016) 579–588. doi:10.1109/TVCG.2015.2466992.

[9] A. Artero, M. de Oliveira, H. Levkowitz, Uncovering clusters in crowded parallel coordinates visualizations, in: IEEE Symposium on Information Visualization, 2004, pp. 81–88. doi:10.1109/INFVIS.2004.68.

[10] G. Andrienko, N. Andrienko, Parallel coordinates for exploring properties of subsets, in: Proceedings. 2nd International Conference on Coordinated and Multiple Views in Exploratory Visualization, 2004, pp. 93–104. doi:10.1109/CMV.2004.1319530.

[11] J. Heinrich., Y. Luo., A. E. Kirkpatrick., D. Weiskopf., Evaluation of a bundling technique for parallel coordinates, in: Proceedings of the International Conference on Computer Graphics Theory and Applications and International Conference on Information Visualization Theory and Applications - IVAPP, SciTePress, 2012, pp. 594–602. doi:10.5220/0003821205940602.

[12] Y.-H. Fua, M. Ward, E. Rundensteiner, Hierarchical parallel coordinates for exploration of large datasets, in: Proceedings Visualization '99 (Cat. No.99CB37067), 1999, pp. 43–508. doi:10.1109/VISUAL.1999.809866.

[13] R. Rosenbaum, J. Zhi, B. Hamann, Progressive parallel coordinates, in: 2012 IEEE Pacific Visualization Symposium, 2012, pp. 25–32. doi:10.1109/PacificVis.2012.6183570.

[14] H. Siirtola, Direct manipulation of parallel coordinates, in: 2000 IEEE Conference on Information Visualization. An International Conference on Computer Visualization and Graphics, 2000, pp. 373–378. doi:10.1109/IV.2000.859784.

[15] J. Blaas, C. Botha, F. Post, Extensions of parallel coordinates for interactive exploration of large multi-timepoint data sets, IEEE Transactions on Visualization and Computer Graphics 14 (6) (2008) 1436–1451. doi:10.1109/TVCG.2008.131.

[16] R. G. Raidou, M. Eisemann, M. Breeuwer, E. Eisemann, A. Vilanova, Orientation-enhanced parallel coordinate plots, IEEE Transactions on Visualization and Computer Graphics 22 (1) (2016) 589–598. doi:10.1109/TVCG.2015.2467872.

[17] R. C. Roberts, R. S. Laramee, G. A. Smith, P. Brookes, T. D'Cruze, Smart brushing for parallel coordinates, IEEE Transactions on Visualization and Computer Graphics 25 (3) (2019) 1575–1590. doi:10.1109/TVCG.2018.2808969.

[18] G. Ellis, A. Dix, Enabling automatic clutter reduction in parallel coordinate plots, IEEE Transactions on Visualization and Computer Graphics 12 (5) (2006) 717–724. doi:10.1109/TVCG.2006.138.

[19] J. Johansson, P. Ljung, M. Jern, M. Cooper, Revealing structure in visualizations of dense 2D and 3D parallel coordinates, Information Visualization 5 (2) (2006) 125–136. doi:10.1057/palgrave.ivs.9500117.

[20] M. Blumenschein, X. Zhang, D. Pomerenke, D. A. Keim, J. Fuchs, Evaluating reordering strategies for cluster identification in parallel coordinates, in: Computer Graphics Forum, Vol. 39 (3), Wiley Online Library, 2020, pp. 537–549. doi:10.1111/cgf.14000.

[21] K. T. McDonnell, K. Mueller, Illustrative parallel coordinates, in: Computer Graphics Forum, Vol. 27, 3, Wiley Online Library, 2008, pp. 1031–1038. doi:10.1111/j.1467-8659.2008.01239.x.

[22] G. Palmas, M. Bachynskyi, A. Oulasvirta, H. P. Seidel, T. Weinkauf, An edge-bundling layout for interactive parallel coordinates, in: IEEE Pacific Visualization Symposium, IEEE, 2014, pp. 57–64. doi:10.1109/PacificVis.2014.40.

[23] R. S. Divino, C. G. Santos, B. S. Meiguins, A visual representation of clusters characteristics using edge bundling for parallel coordinates, in: 2017 21st International Conference Information Visualisation (IV), 2017, pp. 90–95. doi:10.1109/iV.2017.29.

[24] A. Lhuillier, C. Hurter, A. Telea, State of the art in edge and trail bundling techniques, in: Computer Graphics Forum, Vol. 36 (3), Wiley Online Library, 2017, pp. 619–645. doi:10.1111/cgf.13213.

[25] D. Pomerenke, F. L. Dennig, D. A. Keim, J. Fuchs, M. Blumenschein, Slope-dependent rendering of parallel coordinates to reduce density distortion and ghost clusters, in: 2019 IEEE Visualization Conference (VIS), IEEE, 2019, pp. 86–90. doi:10.1109/VISUAL.2019.8933706.

[26] G. Palmas, T. Weinkauf, Space Bundling for Continuous Parallel Coordinates, in: EuroVis 2016 - Short Papers, The Eurographics Association, 2016, pp. 61–65. doi:10.2312/eurovisshort.20161162.

[27] H. Zhou, X. Yuan, H. Qu, W. Cui, B. Chen, Visual Clustering in Parallel Coordinates, Computer Graphics Forum 27 (3) (2008) 1047–1054. doi:10.1111/j.1467-8659.2008.01241.x.

[28] Demonstration video (2022).
URL https://vimeo.com/652208042

[29] RAMP VIS: Visualization and visual analytics in support of rapid assistance in modelling the pandemic (ramp). (2020).
URL https://sites.google.com/view/rampvis

[30] The Scottish Covid-19 Response Consortium: Home page. (2022).
URL https://www.gla.ac.uk/research/az/scrc/

[31] H.-Y. Wu, Y. Niibe, K. Watanabe, S. Takahashi, M. Uemura, I. Fujishiro, Making many-to-many parallel coordinate plots scalable by asymmetric biclustering, in: 2017 IEEE Pacific Visualization Symposium (PacificVis), 2017, pp. 305–309. doi:10.1109/PACIFICVIS.2017.8031609.

[32] M. Lind, J. Johansson, M. Cooper, Many-to-many relational parallel coordinates displays, in: 2009 13th International Conference Information Visualisation, 2009, pp. 25–31. doi:10.1109/IV.2009.43.

[33] K. Koc, A. S. McGough, S. Johansson Fernstad, Peaglyph: Glyph design for investigation of balanced data structures, Information Visualization 21 (1) (2022) 74–92. doi:10.1177/14738716211050602.

[34] J. Fuchs, P. Isenberg, A. Bezerianos, D. Keim, A systematic review of experimental studies on data glyphs, IEEE Transactions on Visualization and Computer Graphics 23 (7) (2016) 1863–1879. doi:10.1109/TVCG.2016.2549018.

[35] R. Borgo, J. Kehrer, D. H. S. Chung, E. Maguire, R. S. Laramee, H. Hauser, M. Ward, M. Chen, Glyph-based Visualization: Foundations, Design Guidelines, Techniques and Applications, in: Eurographics 2013 - State of the Art Reports, The Eurographics Association, 2013, pp. 39–63. doi:10.2312/conf/EG2013/stars/039-063.

[36] R. C. Roberts, L. McNabb, N. AlHarbi, R. S. Laramee, Spectrum: a C++ header library for colour map management, in: Proceedings of the Conference on Computer Graphics & Visual Computing (CGVC), Eurographics Association, 2018, pp. 135–141. doi:10.2312/cgvc.20181218.

[37] K. A. Bollen, K. H. Barb, Pearson's r and coarsely categorized measures, American Sociological Review (1981) 232–239doi:10.2307/2094981.

[38] T. Hogan, U. Hinrichs, E. Hornecker, The elicitation interview technique: Capturing people's experiences of data representations, IEEE Transactions on Visualization and Computer Graphics 22 (12) (2016) 2579–2593. doi:10.1109/TVCG.2015.2511718.

[39] Covid-19 EERA Model (2022).
URL https://github.com/ScottishCovidResponse

APPENDIX: Covid-19 Simulation Data

The simulation model used is from the Epidemiology, Economics and Risk Assessment (EERA) model [39]. The model incorporates an inference process to estimate the range of parameters of interest and the ranges of parameters to extract parameter configurations. In this case, there are 160 parameter configurations. For each configuration there are multiple simulation runs. In this case, 1000 runs result in different predictions.

The model takes the same set of input parameters, called simulation configurations that yield different output results for each run. The model aims to provide the range of output possibilities for each possible prediction. For each output result, minimum, maximum and mean values of output parameters are provided.

For the model, there is a long list of parameters, some are inferred, some are estimated a priori, and some are fixed across runs. Here are the critical input parameters:

- **nsse_cases:** Normalised sum of square error for the number of cases
- **nsse_deaths:** Normalised sum of square error for the number of deaths
- **p_inf:** Probability of infection
- **p_hcw:** Probability of infection (Healthcare worker)
- **c_hcw:** Mean number of healthcare worker contacts per day
- **d:** Proportion of population observing social distancing
- **q:** Proportion of normal contact made by people self-isolating
- **p_s:** Age-dependent probability of developing symptoms
- **rrd:** Risk of death if not hospitalized
- **lambda:** Background transmission rate

For each age group (8 age groups) there are;

- 200 days of predicted time-series of each output data dimension in the model
- 16 distinct output data dimensions (see the list below)

The model generates a number of output files for each run. In total, 160 (parameter configurations) $\times$ 16 (data dimensions) $\times$ 1000 (runs) $\times$ 8 (age groups) = 20,480,000 time series of 200 days each. The data we display by default is the first configuration.

The output simulation parameters are as follows:

- **Age Group:** Age groups ID are used in the model.
- **Day:** The day for the record
- **S:** Number of susceptible individuals (not infected)
- **E:** Number of infected individuals but not yet infectious (exposed)
- **I:** Number of infected and infectious asymptomatic individuals
- **IS:** Number of infected and infectious symptomatic individuals
- **H:** Number of infected individuals that are hospitalized
- **R:** Number of infected individuals that are recovered from infection
- **D:** Number of dead individuals due to disease

The age groups ID as used in the model are here:

- **Group 1:** Under 20
- **Group 2:** 20-29
- **Group 3:** 30-39
- **Group 4:** 40-49
- **Group 5:** 50-59
- **Group 6:** 60-69
- **Group 7:** 70+
- **Group 8:** Health Care Workers