

# Graphical Enhancements for Effective Exemplar Identification in Contextual Data Visualizations

Xinyu Zhang, Shenghui Cheng\*, and Klaus Mueller, *Senior Member, IEEE*

**Abstract**—An exemplar is an entity that represents a desirable instance in a multi-attribute configuration space. It offers certain strengths in some of its attributes without unduly compromising the strengths in other attributes. Exemplars are frequently sought after in real life applications, such as systems engineering, investment banking, drug advisory, product marketing and many others. We study a specific method for the visualization of multi-attribute configuration spaces, the Data Context Map (DCM), for its capacity in enabling users to identify proper exemplars. The DCM produces a 2D embedding where users can view the data objects in the context of the data attributes. We ask whether certain graphical enhancements can aid users to gain a better understanding of the attribute-wise tradeoffs and so select better exemplar sets. We conducted several user studies for three different graphical designs, namely iso-contour, value-shaded topographic rendering and terrain topographic rendering, and compare these with a baseline DCM display. As a benchmark we use an exemplar set generated via Pareto optimization which has similar goals but unlike humans can operate in the native high-dimensional data space. Our study finds that the two topographic maps are statistically superior to both the iso-contour and the DCM baseline display.

**Index Terms**—High-dimensional data, multivariate data, contextual displays, exemplar generation, decision support, configuration space

## 1 INTRODUCTION

WE address the task of visually selecting a set of exemplars from a larger set of data objects with multivariate characteristics. An exemplar set is a compact set of data objects where each balances the tradeoffs that exist among the characteristics in a unique way. To illustrate where this can be useful consider a wine shop that has only limited shelf space but still wishes to satisfy a broad customer base. The owner will seek to stock a set of wines that minimizes redundancies and at the same time offers a well-diversified spectrum that vibes well with most buyers but is devoid of extreme selections that only appeal to niche customers. These types of challenges occur in many application areas, such as investment banking where balancing a stock portfolio is a key to solid growth, in drug design where components have different side effects that affect different patient populations, and even in amusements parks where different attractions will appeal to different audiences, just to name a few. All of these will want to choose a limited set of configurations where each of these is unique in its combination of strengths and weaknesses.

Identifying good exemplar sets can be taxing. It requires one to recognize interactions that exist among the attributes and then observe how these interactions are expressed in the available data objects. When these interactions are spread over multiple bivariate tiles, as in Scatterplot Matrices [27], or spread across multiple axes, as in Parallel Coordinates [33], discerning them becomes a difficult undertaking. While embeddings like MDS [38], t-SNE [62], or UMAP [43] present a more concerted view, they do not embed information that relates the data objects to the attributes, prohibiting a user from appreciating the trade-offs that guide the

selections.

What is needed is a visualization that can integrate the data objects and the attributes that describe them into a single display. We call these visualizations *Attribute-Contextualized Data Visualizations*. A prominent visualization of this genre is the Data Context Map [13]. It embeds the data objects as points into a map of attribute nodes where the relative proximity of a data object to an attribute node denotes the attribute's strength in the data object's overall characteristics profile, that is, the closer a data object is to an attribute node, the more pronounced the attribute's influence tends to be. Likewise, similar data objects locate in close neighborhoods and so do the nodes of correlated attributes. Fig 3(a) shows an example for an automobile dataset.

The non-linear warp that all embedding techniques apply when mapping the high-D data space into 2D leads to local distortions. This has been well studied in recent years [20]. For the DCM it means that the iso-value contours around an attribute node are typically not concentric circles but more general iso-contours, with their density indicating the local space warping. Fig. 3(c) shows these contours for the HPower attribute of the automobile dataset.

### 1.1 Preference Sets vs. Exemplar Sets

The DCM as described in [13] was conceived for a Preference Set Selection task. It was designed in such a way that it could tolerate these embedding distortions. Users would begin by manipulating a set of sliders to specify the desired value range for each attribute. Moving a slider then changed the extent of a filled region on the map, where each such region was bounded by the iso-contours defined by the specified slider value (the closed contours in Fig. 3(c)). The regions so created then contained the set of data configurations (blue points in Fig. 3(c)) that fit the specified value brackets set by the associated attribute slider. Finally, map areas where the regions of several attributes overlapped contained the configurations that fit the multiple selection criteria of interest. Users could then further play with the sliders to fine-tune the set of configurations that had the preferable multivariate characteristics.

• X. Zhang and K. Mueller are with the Department of Computer Science, Stony Brook University, NY.  
E-mail: {zhang146, mueller}@cs.stonybrook.edu

• S. Cheng is with WestLake University and Westlake Institute for Advanced Study, Hangzhou, China  
E-mail: chengshenghui@westlake.edu.cn (corresponding author)

Manuscript received August 10, 2021; revised ....

The current task of Exemplar Set Selection is fundamentally different from the aforementioned task of Preference Set Selection. It is a much broader task since the set of exemplars are chosen to represent compromises for *multiple* preference sets. With the method of [13] each such exemplar would first require the specification of a contour overlap region followed by an extraction from the covered set of configurations. Doing this for every exemplar would be a highly laborious and likely inaccurate undertaking.

Visual exemplar identification is essentially a map localization task. Users need to be able to look at the map and point out these landmarks by assessing their (attribute node) surroundings. In fact, exemplars are a good localization target since they represent the most salient tradeoff goal posts in a configuration space. Then, once an exemplar is localized, users can inspect its immediate neighborhood to engage in further personal optimizations.

To make this localization task easier in the presence of the aforementioned map distortions, we have devised several graphical enhancements which we have studied in this paper. While exemplars could also be found automatically via Pareto optimization, the returned options may not always be practical or even feasible in a specific application scenario, especially when many variables are involved. Affording the expert more control is thus desirable. Further, the graphical enhancements we have designed not only enable users to locate exemplars; rather, they also elevate the potential of endowing users with a better appreciation of the configuration space in general.

## 1.2 Goals and Contributions

The goals and contributions of our paper are as follows:

- A visual interface that enables users to select a set of representative exemplars from a high-D multi-item dataset.
- Framing this task as a high-D space localization task and adapting for it an existing Attribute-Contextualized Data Visualization, the Data Context Map (DCM).
- A set of graphical enhancements for the DCM purposed to better convey the global trade-offs among the data features.
- A comparative methodology that can test if these enhancements can effectively facilitate high-D localization tasks, using exemplar set selection as a clearly defined task.
- An enhanced Pareto frontier algorithm that is aware of the specific goal of exemplar set selection and can be used to derive suitable benchmark data.
- A user study to test and compare the developed designs in terms of these goals.

The structure of our paper is as follows. Section 2 presents related work. Section 3 provides background on the DCM. Section 4 introduces the graphical enhancements we devised. Section 5 describes our user interface. Section 6 presents our enhanced Pareto frontier algorithm used to derive benchmark data. Section 7 details our user study and presents results. Section 8 ends with conclusions.

## 2 RELATED WORK

In this section we will primarily focus on two main areas related to the visualization of multivariate data: (1) methods that produce a single, joint display of data and attributes (the aforementioned Attribute-Contextualized Data Visualizations), and (2) methods that seek to facilitate Multi-Objective Optimization (MOO) since finding a set of exemplars is related to MOO. Related to the first set are methods that invoke visual representations commonly used in topographic maps. We will summarize this work here as well.

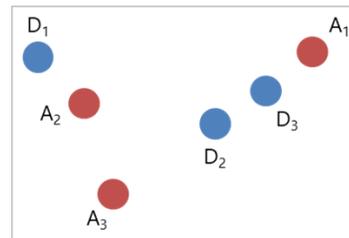


Fig. 1: Attribute-Contextualized Data Visualization concept. The red nodes are the attributes and the blue nodes are the data points.

### 2.1 Attribute-Contextualized Data Visualizations

We define Attribute-Contextualized Data Visualizations as methods that display data points in relation to the data attributes in a joint layout. When viewing such a visualization an analyst can make three types of assessments: (1) how similar a data point is to another, (2) how correlated an attribute is with another; and (3) how strongly expressed an attribute is for a given data point. Fig. 1 illustrates these three properties; the red nodes are three attributes, say the scores for a three-subject exam testing a student’s knowledge of English ( $A_1$ ), Math ( $A_2$ ), and Arts ( $A_3$ ) while the blue nodes ( $D_1$ ,  $D_2$ , and  $D_3$ ) represent three students who took this three-subject exam. We quickly see that  $D_1$  has a strong affinity for Math, but is weak in the other two subjects and somewhat of an outlier. We also see that  $D_2$  and  $D_3$  are more alike, with  $D_2$  being an all-rounder while  $D_3$ ’s strength is more inclined to English.

Popular multivariate visualization methods like Parallel Coordinates (PC) [33] or Scatterplot Matrices (SM) [27] do not simultaneously support all of these three assessments. For PC, attribute correlations are difficult to discern unless the two attribute axes are adjacent, while for SM, data object similarities are difficult to see since each bivariate tile conveys this similarity only for two attributes at a time. Finally, embedding methods like MDS [38], t-SNE [62] and UMAP [43] do not maintain the attributes at all.

There are several methods that can simultaneously support all three assessments. RadViz [29] positions the attribute nodes on a circle and maps the data points into the circle’s interior. A similar layout is also produced by Gravi++ [28]. The VIBE system [51] follows a similar concept for the visualization of documents with respect to certain keywords. Star Coordinates [34] are similar to RadViz but they do not perform a prior normalization and so the data points are not confined to a bounded area. Dust+Magnet [59] generalizes these methods by allowing users to move the attributes nodes to any position and so change the layout of the data points. StarPlots [10] are a radial version of Parallel Coordinates. Radar Charts [2] are similar to StarPlots but typically fill the area within the closed contour. All of these visualizations produce very different point arrangements when the order of attributes or their polar spacing is changed. This variability arises since the mapping of the multivariate data into the 2D surface is linear which might disregard some of the data variances and so cause dissimilar data points to be mapped close to one another. Optimization of the attribute ordering and spacing can yield less ambiguous layouts. This can be further improved by applying optimization techniques gleaned from non-linear embedding techniques. RadViz Deluxe [14] achieves this by non-linearly optimizing the visual mappings among (1) the data points and (2) the data points and the attributes. It also uses correlation-based attribute ordering and spacing. Recent work that goes into a similar direction is RadViz++ [53].

The DCM generalizes the placement of the attribute nodes from the RadViz circle boundary into the data point layout. This is achieved by a fusion of the data similarity matrix and the attribute correlation matrix followed by an MDS-type embedding optimization; see Section 3 for a more detailed discussion. The generalization can reduce layout ambiguities further and also allows the depiction of iso-contours around the attribute nodes to visualize how iso-value hyperspheres around attributes are distorted when mapped to a 2D canvas. The contours can give important visual explanations for the relative strengths of data points with respect to the attributes. The concept of iso-contours for the visualization of space distortion was also exploited in DimReader [20] which generates the iso-contours via the perturbation of input points, but this work does not visualize the attributes as a contextualization. Work more related to the DCM is the method by Broeksema et al. [8] which uses Multiple Correspondence Analysis (MCA) to derive a joint object–attribute layout and then defines topical regions from these. But MCA only applies to categorical data while the DCM also applies to numerical data. In some sense, their method is also related to the correlation map by Zhang et al. [70] which tessellates an attribute correlation network and then projects the data into these tiles by generalized barycentric interpolation [44].

## 2.2 Visual Support for Multi-Objective Optimization

The identification of exemplars is related to the general field of multi-objective/multi-criteria decision making (MCDM). MCDM has its roots in operations research; it deals with the problem of assessing trade-offs under multiple conflicting criteria and picking out the set of configurations that provide a good balance [36], [66]. MCDM can get exceedingly difficult even with small configuration spaces spanned by just a handful of parameters. Just think of the common task of buying a car, selecting a college, or picking out a bottle of wine. There are usually not that many criteria but it is nevertheless often difficult to narrow down the choices to a few hot list candidates (the exemplars) where the choice task is less overwhelming. While the above choice tasks are more personal, similar problems also occur in “larger” settings, such as healthcare [52], environmental management [35], and many others.

In seminal work, Miller [46] demonstrated that humans find it difficult to deal with more than 7 pieces of information (plus/minus 2) at the same time. It is therefore advisable to narrow down the number of information items used in an active human-based decision making task like exemplar identification to a manageable subset, for example via dimension reduction or interactive semantic steering [41]. Miller further showed that via “chunking” humans can deal with a much larger number of items. In visualization, clustering and grouping can be seen as a form of “chunking”. Our methodology assumes that such a narrowing has taken place before the visual analytics process commences. Most papers in the research field of visual analytics also showcase their methods with these limited, but human-manageable information spaces.

One way to deal with MCDM tasks is to ask the user to define weights for each of the parameters followed by a ranking of the configurations based on the weighted sum of parameters. The LineUp system by Gratzl et al. [23] enables users to create these rankings interactively by simply modifying the weights via sliders, while the WeightLifter system by Pajer et al. [54] enables users to delve deeper into the weight space via a brushable parallel coordinate plot. Dimara et al. [18] evaluate the capacity of several basic visualization paradigms, such as parallel coordinates, scatterplot matrices, and tabular visualizations to support MCDM.

A more automated way to determine exemplars and help in MCDM is Pareto optimization [31], [45]. Pareto optimization seeks to locate the set of configurations at which the value gauging one of the criteria can be improved without degrading the values tied to one or more competing criteria. There can be a potentially large set of such Pareto optimal configurations, especially when the configuration space has many dimensions. To aid analyst in navigating the Pareto solution space several methods have been devised, some of which are visual, while others are more analytical.

Lotov et al. [40] offer two approaches to visualize the Pareto optimal set. For convex sets, they suggest a representation as convex polyhedra which can be visualized as a decision map, while for non-convex sets they use a point-wise approximation which can be visualized as a heatmap [67] or a scatterplot [22]. Berezkin et al. [5] propose an adaptive method to approximate the Pareto set in the non-convex case. Veldhuizen et al. [65] describe an evolutionary search algorithm and demonstrate its convergence to the Pareto frontier, while Abbass et al. [4] describe a PDE-based algorithm to optimize problems over continuous domains using differential evolution. Chen et al. [11] discuss how searching the Pareto set in a distributed manner can speed up the process. It is noteworthy that while these methods can converge to a valid result fairly quickly, a Pareto set can be unstable even for small parameter value changes [57]. These variations can then result in substantially different Pareto sets. Our method addresses this problem by enabling users to explore what effect a change in the value ranges has on the Pareto set, and hence observe the set’s sensitivity to these changes.

Besides these more traditional visualization methods, Blasco et al. [6] contribute the concept of Level Diagram to analyze high dimensional Pareto sets by layer classification and synchronize representations of all data objects and parameters. Nasrolahzadeh et al. [49] propose Pareto-RadVis which projects data objects to a radial coordinate plot to show the ranks of the Pareto set, the relative locations of data objects and the candidate solution distributions. Yet, it remains difficult for decision makers to easily express their own preferences and constraints on the decision variables and obtain a solution set interactively. Trade-off decisions are also not clearly reflected. In contrast, our system allows interactive exemplar selection along multiple perspectives with immediate visual feedback. It presents a multivariate Pareto frontier as a piecewise linear path across a 2D decision map where each Pareto optimal solution is connected in a specific similarity order. In this way decision makers can serialize the trade-offs they make according to the order of the path and so read them like a storybook.

The method that in spirit is closest to ours is the one by Chen et al. [12]. They propose a semantically enhanced Self Organizing Map (SOM) to encode Pareto sets in a 2D polygon. Decision makers can define their preferences interactively and obtain a candidate solution from the clustered results. However, the variables all reside on polygon corners and as such have only limited ability to reveal how data objects and variables influence each other. As mentioned, our method intermingles attributes and data points into a shared canvas which has the advantage of revealing association between data objects and variables.

Finally, another MOO application is robot path planning. It aims to find the best path from start to destination based on factors like path length, time, environmental obstacles, computational complexity, etc. A good path planning algorithm should make a tradeoff among conflicting factors. Duchoň et al. [19] introduced some path planning algorithms for mobile robot like the A\* algorithm [26], Phi\* algorithm [48] and JPS [25]. Lavin further

combined A\* search and Pareto optimization [39] for path planning to find a better path. Ferariu et al. [21] proposed a self-adaptive algorithm to monitor the size of the Pareto front during path planning. Our Pareto path was inspired by Lavin’s method.

### 2.3 Visualizations Inspired by Topographic Maps

While the methods discussed in Section 2.1 projected the high dimensional data into a 2D plane, topographic methods encode additional information, such as the density or number of data items, into 2D contours or a 3D elevated surface. These are then graphically represented in ways akin to a typical topographic map. A well-known paradigm here is ThemeScapes [69] which has found many uses for the theme-organized visualization of large document collections, such as books, papers, and patents.

Hogräfer et al. [30] studied a set of typical map-like visualization technologies. These technologies were classified from two perspectives, imitation and schematization. Imitation describes technologies that use features of a map in a visualization scheme for non-spatial data (e.g. [24]) while schematization summarizes technologies that transform geographic maps to visualize more thematic information for spatial data (e.g. [63]). Imitation and schematization describe a continuum of visualization and map. Therefore, non-spatial data and spatial data were summarized in a unified context of map-like characteristics.

Tory et al. [60], [61] compared the performance of 2D landscape, 3D landscape and dot display. Here, non-spatial data was visualized as a plane, additional information like special attribute or classification tag [9], [69] was mapped to height which derived a 2D or 3D landscape belonging to the imitation type discussed above. They tested these visualization schemes with different tasks: estimating the number of points of a specified color within a spatial area or memorizing the spatial distribution of a dataset. Their results showed that landscape visualizations did not outperform the dot display in either of the tasks. However in more complicated tasks, like finding Pareto optimal points or searching points that satisfy given attribute constraints, how landscapes perform has not been evaluated thus far.

Similar to landscapes, Kraus et al. [37] carried out an empirical study on 2D and 3D heatmap performance. They found that in reading and comparison tasks, 3D heatmaps were better in terms of error rate, but well-established 2D heatmaps outperformed 3D heatmaps in overview tasks. Although 3D visualization was not superior to 2D as one might have expected, Marriott et al. [42] discussed some potential applications of 3D in information visualization, including using 2.5D to show an additional data dimension which is part of our system.

## 3 BACKGROUND: THE DATA CONTEXT MAP

The Data Context Map (DCM) [13] is a typical Attribute-contextualized data visualization. We choose it to examine our graphical enhancements. In this section we give a brief introduction to it. The DCM achieves a composite display by fusing together four types of matrices. Assume a rectangular data matrix  $DM$  of  $m$  data points and  $n$  attributes which is first normalized into a  $[0, 1]$  interval. The DCM creates four matrices that are fused together into an  $(m+n) \times (m+n)$  joint matrix. To illustrate consider Fig. 2(a) which shows the joint space obtained by simultaneously mapping the data points into a space spanned by the attribute vectors and mapping the attributes into a space spanned by the data vectors. One space is obtained from the other by transposing the  $DM$ .

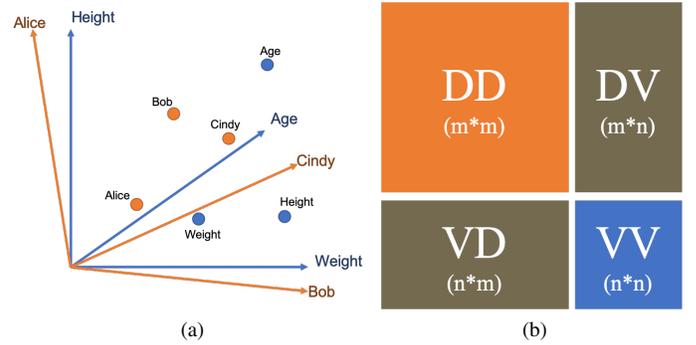


Fig. 2: An illustration of the DCM matrix fusion process using a simple demographics dataset with three individuals (Bob, Cindy, Alice) and three attributes (age, weight, height). (a) The data space and the attribute space are fused together (b) The fused matrix.

The fused matrix is shown in Fig. 2(b). The matrix  $DD$  is the  $m \times m$  distance matrix which holds the pairwise Euclidean distances of the  $m$  data points. The matrix  $VV$  is the  $n \times n$  ( $1 - correlation$ ) matrix which gauges the similarity of the attributes. The remaining two matrices  $DV$  and  $VD$  relate each of the  $m$  data points to each of the  $n$  attributes and vice versa. For the fused matrix to be symmetric the only similarity metric that can fulfill this condition is  $(1 - value)$ , where  $value$  is the value a data point has for a particular attribute. The  $(1 - value)$  distance can be thought of as a significance distance. It is small for a given data point when the value of a point’s attribute is large, encoding a notion of affinity that the data point has for the attribute. Using this metric makes  $VD$  a transpose of  $DV$  and vice versa.

Once the fused matrix has been obtained it can be embedded via MDS (we use Glimmer MDS [32]). Fig. 3(a) shows such an embedding for the car dataset [1] where the red nodes are attribute points and the blue nodes are the data points, the cars. In this figure we shaded each data point by its value in the horsepower attribute. We observe that the data points fade out towards the top of the image but that the gradient is not linear. Next adaptive kernel density estimation (AKDE) [64] is applied to interpolate the space between the data points and so generate a continuous value visualization. AKDE is preferred over standard KDE since it uses smaller kernels in dense neighborhoods and larger kernels in sparse neighborhoods reducing blur in dense regions and gaps in sparse regions. Fig. 3(b) shows the AKDE-filtered value field of (a).

The final step is the generation of iso-contours via the CONREC algorithm [7] (see Fig. 3(c)). The DCM [13] enables users to interactively select specific iso-contours, one per attribute, via an array of value sliders. Data points that meet or exceed a selected slider value will then fall within the region inscribed by the slider attribute’s iso-contour. The compound visualization enables users to visually assess the trade-offs associated with different slider settings and eventually arrive at a set of points that meet all criteria. We note, however, that this set of points are not exemplars; rather they are simply points that meet all or some selection criteria.

## 4 GRAPHICAL ENHANCEMENTS

As mentioned, the goal of this work was to test whether a graphical enhancement of the DCM can aid the understanding of trade-offs in multivariate configuration spaces. For this purpose we chose to test a more challenging task, namely, the identification of exemplars. We studied three enhancements (and the reference visualization),

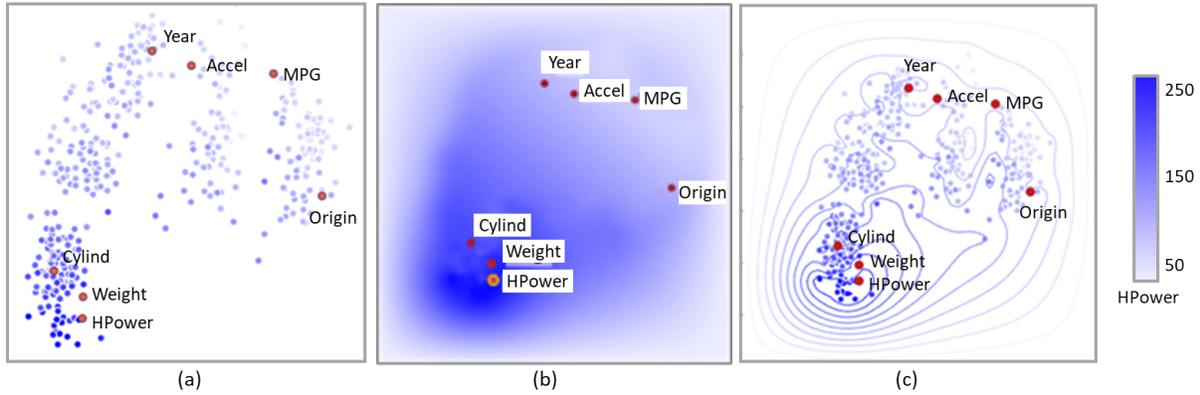


Fig. 3: Building the contoured Data Context Map, shown here is the car dataset. (a) The basic MDS layout with nodes for data points (blue) and attributes (red); the data points are shaded by their values in the horsepower (HPower) attribute (bottom left). (b) The map of (a) interpolated via AKDE. (c) The continuous map of (b) with iso-contours drawn.

all of which are representative of enhancements commonly used for maps. Fig. 4 shows all of these visualizations for the car dataset where panel (a) is the original data context map which serves as the reference visualization since it has no graphical enhancements at all. In the following we describe the three enhancements we tested.

#### 4.1 Single Iso-Contour

This enhancement, shown in Fig. 4(b) was already part of the original DCM. In this visualization constraints are represented by colored areas where data configurations that fall into these areas satisfy the corresponding criteria. Each contour visualizes a particular attribute constraint. The analyst can learn how the various constraints distribute spatially and see for each data point which of the constraints it fulfills by containment in a region. However, since the contours can have irregular shapes, actual values can not be easily discerned. A user can tighten or loosen an attribute constraint by adjusting the iso-contours via a slider, but it can be a tedious effort to learn about sensitivities that way, especially for multivariate data. Furthermore, for larger sets of attributes there will be many areas with different colors which can lead to cognitive overload, particularly when the colored areas blend and overlap. Hence, while the iso-contoured DCM is well suited to a determined candidate set that fits certain constraints, it is less suited to pin down an exemplar set that balances tradeoffs.

#### 4.2 Value-Shaded Topographic Rendering

Fig. 3(c) shows a display that resembles a topographic map. It shows a set of iso-contours for a specific variable (here horsepower) making it easy to gauge local value sensitivities. But we found that when iso-contour fields for different attributes overlapped these sensitivities were difficult to discern, even when the iso-contours for each attribute were assigned a different color. To address this problem, we returned to the filled iso-contours of the original DCM. But instead of using a constant color for the entire constraint value range, we used progressively darker colors for each iso-contour region and blended the filled shapes together.

The outcome is shown in Fig. 4(c). We found that this works quite well with limited number of colors (we found the limit is 4-5 attributes by experience). The light to dark color gradients of the filled iso-contours visualize the increasing constraint levels while the mixing of colors from adjacent attributes aid in the recognition of tradeoffs, but blending more than 5 colors can lead to issues of interpretability.

#### 4.3 Topographic Terrain Rendering

A topographic terrain rendering shows a geographic map with additional shading to convey the height of a landscape. Terrain rendering is common in geography, but in visualization 3D rendering schemes (e.g. [55], [56]) were frequently argued not to provide additional cognitive benefits [16], [17], although many viewers find them more appealing. The most significant shortcomings of 3D information displays are occlusion and perspective distortion.

Encouraged by other work that showed that terrain rendering can be made efficient for information visualization [50], we implemented a 2.5D terrain rendering scheme; see Fig. 4(d) for the car dataset. We chose colormap, shading and viewing angle to emphasize height and so avoid problems related to occlusion and perspective. The contours are drawn similar to the topographic map but with one major difference. While the topographic map constructs a set of contours for each attribute individually and then superimposes the emerging shapes, the terrain map sums all attribute values first and then constructs the contours. In this way only a single terrain emerges. The upside of this scheme is that ambiguities due to blending are avoided. While this restricts visual access to the individual strengths of the attributes in a certain map area, it replaces it with the combined strength related to these attributes. It affords the user a pre-computed local optimization which they had to do mentally with the blended topographic display.

### 5 USER INTERFACE

We have constructed a user interface that enables users to (1) define a desirable range of values for each attribute using data-scented sliders [68], (2) select a set of exemplars in the graphically-enhanced DCM visualization, and (3) observe the trade-offs within this set of exemplars in a stacked area chart. We allow users to select allowable value ranges first to limit the graphical enhancements to the relevant areas of the map. This focuses the user's attention and so reduces cognitive load. Fig. 5 shows the user interface elements for a dataset of Pokémon characters [3].

Fig. 5(a) shows the value sliders which are scented by a continuous histogram to inform the user how many data instances are within a selected interval. We automatically subtract data instances that are culled out by other sliders. The lower value limit for each attribute will form the outer-most contour in the DCM for the respective attribute. We always associate the right-most value with the most-attractive value setting. This is needed

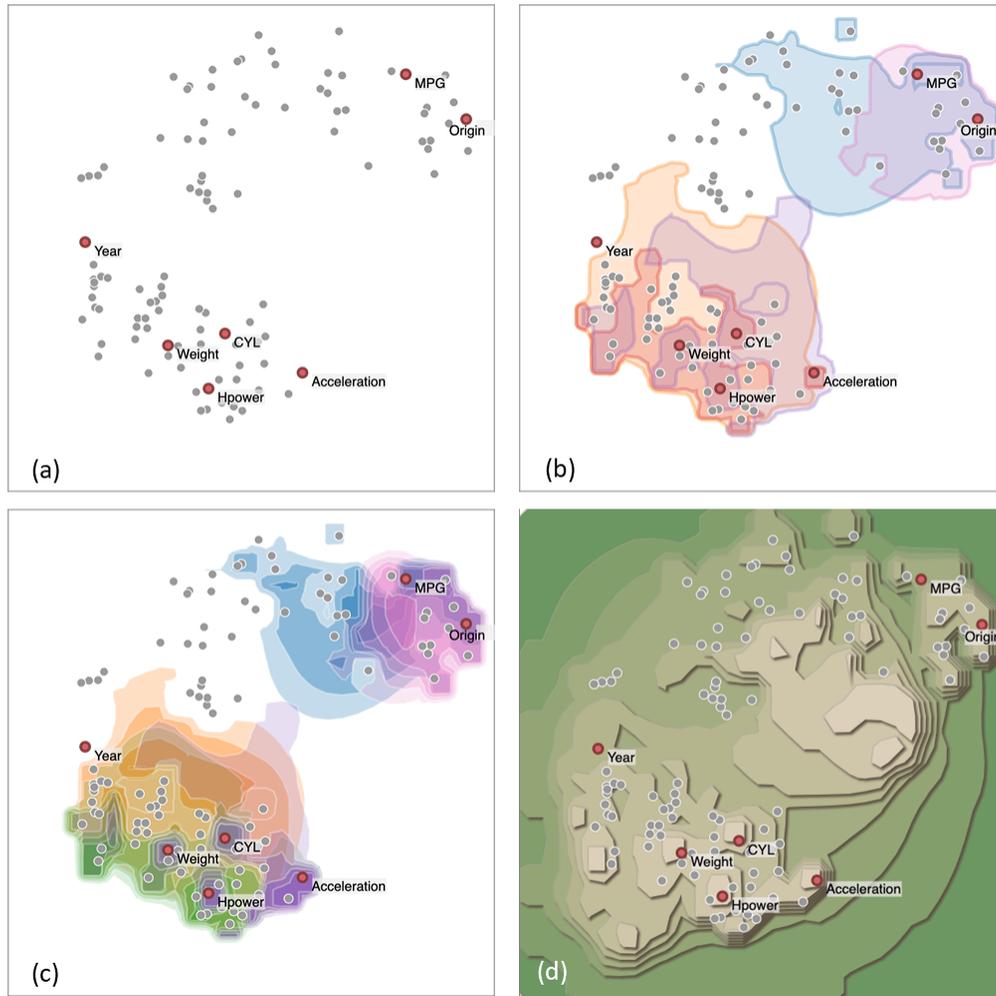


Fig. 4: The three graphical enhancements we studied. (a) The original data context map functioning as the reference visualization, (b) the contour map used in the interactive data context map, (c) the value-shaded topographic map, and (d) the topographic terrain map.

to ensure that attribute values are mapped to height or shade in a semantically consistent fashion. For example, for an attribute like ‘price’ a lower value will be more attractive. We therefore convert ‘price’ to ‘affordability’ using the transform  $(1 - \text{price})$  after the  $[0, 1]$  normalization step. Other examples are ‘time to 60 MPH’ which one can map to ‘acceleration’ by using the reciprocal.

Fig. 5(b) shows a set of user-selected Pokémon character exemplars in the value-shaded topographic map. After selection our program auto-connects these instances by a graph using a shortest path algorithm. This path is meant to be a line-up of exemplars with smoothly changing trade-offs. The stacked area chart shown in Fig. 5(c) visualizes this smooth path with the attribute values mapped to width. The visualization shows how some attributes loose in strength while others gain in strength. Most prominently, the attack and the defense characteristics are inversely related but it is more complicated than that if one also takes into account other traits like ‘speed’ and ‘HP (Hit Points)’ which rates how much damage a Pokémon can receive. In the middle, we can clearly make out Arceus, the most powerful Pokémon who has all traits equally well expressed, with a slight preference for ‘speed’ and ‘HP’.

## 6 GENERATION OF BENCHMARK PARETO PATHS

To evaluate the quality of the user-generated exemplar sets we require a benchmark exemplar set. As mentioned, Pareto opti-

mization is a well-established method to determine configurations that capture the preference criteria we expect from the selected exemplars. However, while in 2D a Pareto frontier is typically a path-like contour that covers the set of configurations, when the dimensionality is greater than 2D this cover generalizes to a manifold. To generate a path that orders the identified Pareto configurations in a similar fashion than the user-generated path shown in Fig. 5(b) we devised an algorithm that mimics the user’s task as close as possible. First, it takes into account only the attribute ranges selected by the scented sliders. Second, it uses the same two extreme configurations also selected by the user; these are the configurations at the opposite ends of the path of exemplars in Fig. 5(b), call them *terminal exemplars*. Our Pareto path algorithm then constructs a piece-wise linear path across the configuration space using an A\* search-based scheme; it starts with one of the terminal exemplars and then iteratively searches among evolving sets of Pareto optimal successors until the other terminal exemplar is reached.

Our method is inspired by the algorithm devised by Lavin [39] who formulates the task of planning an optimal path for a mobile robot as a multi-objective optimization (MOO) problem. MOO is appropriate since apart from just minimizing the start-to-goal distance there are often additional objectives such as cost, safety, and time. To solve this problem Lavin offers an approach that uses

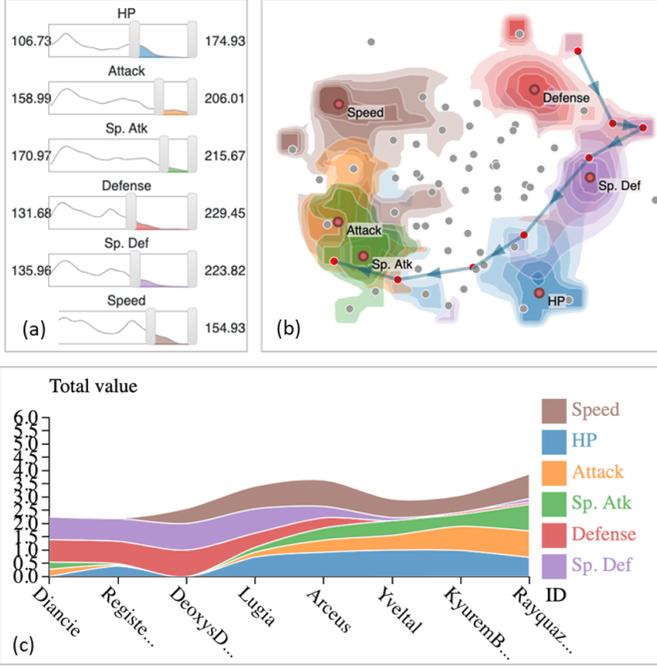


Fig. 5: User interface components, shown here is the Pokémon dataset. (a) Scented slider widgets, one for each of the 6 user-chosen attributes, (b) shaded topographic map of these 6 attributes with the 8 user-marked exemplars auto-connected by a path, and (c) stacked area chart with the 6 attributes and the 8 exemplars. It can be observed that each exemplar has a unique attribute configuration.

A\* search across the Pareto frontier. A robot’s path is a smooth trajectory through the configuration space and so it quite resembles what we are trying to achieve. Yet, since our objective differs somewhat from robot path planning we made certain modifications of this scheme. We present our approach next.

## 6.1 Our Pareto-Guided A\* Search Algorithm

Our first modification is with regards to taking into account the user-chosen slider settings in the Pareto search. In order to satisfy this constraint we use a clone of the data matrix,  $DM'$ , where a cell is transformed to a small value close to zero (like, divided by 20) when the original data value for that cell’s attribute falls outside the selected slider range. This will reduce the attractiveness for that data point in the Pareto optimization but it will not completely eliminate it. Our second modification is to cast a specific set of criteria to decide on the quality of the chosen Pareto configurations. Here we aim for points that are strong in their features but at the same time have smooth transitions in these strengths along the constructed Pareto path.

A\* search embodies a best-first search strategy. It uses a priority queue, called *open list*, to keep a set of possible successors to the current head node  $Q$  on the evolving path. A successor is chosen as the node  $X$  that minimizes the sum  $f(X)$  of the cost of the path from the start node to  $X$ ,  $g(X)$ , and a heuristic function  $h(X)$  that estimates the cost of the cheapest path from  $X$  to the target.

Algorithm 1 shows the pseudocode of our algorithm. The *close list* will eventually contain the path instances and starts out empty; the *open list* starts out with one of the two terminal exemplars. At each step along the path construction we identify, in  $DM'$ , the  $K$  data instances that are most similar to  $Q$  (we found that setting  $K = 12$  worked best) as the  $Q$ .*adjacent* set; similarity is measured

by cosine distance. Given these  $K$  candidate instances we run a Pareto optimization (the *criteria optimization* in Algorithm 1) in terms of the user-chosen attributes to find the Pareto set among the  $K$  candidates.

Using the generated Pareto set we perform a second optimization, a *distance-based Pareto optimization*. It minimizes two distances and a magnitude: (1) the distance  $\|Q_{DCM} - X_{DCM}\|_2$  of the candidate instance  $X$  with  $Q$  on the Data Context Map (DCM), (2) the  $L_1$  norm  $\|\mathbb{1} - X\|_1$  to encourage candidates with strong features, and (3) the cosine distance  $(1 - \cos \langle Q, X \rangle)$  to encourage candidate instances that are close to  $Q$  to enable smooth transitions on the path. We subtracted from 1 in (2) and (3) since we are minimizing the Pareto frontier, and we used the DCM space in (1) since that is where the user will visualize the data. Besides, it is also easier to force the path to make sufficient progress by considering distances in DCM space since point distances are mapped to data relationships in that space.

The candidate instances selected in these two consecutive optimizations then form the list of *successors* in the subsequent A\* search. This search uses the following local cost function:

$$g_0(X) = \|Q_{DCM} - X_{DCM}\|_2 + \|\mathbb{1} - X\|_1 + (1 - \cos \langle Q, X \rangle) \quad (1)$$

where the terms are similar to the terms used in the first stage of the local Pareto search explained above.

We aggregate the winning  $g_0(x)$  at each step during the path evolution and store it as an average in  $g_n(x)$ , where  $n$  is the number of steps along the path so far. Combining these terms then gives the overall cost function  $g(X)$  as follows:

$$\begin{aligned} g(X) &= (g_n * n + g_0(X)) / (n + 1) + 1 / (30 - n) \\ &= (g_n * n + \|Q_{DCM} - X_{DCM}\|_2 \\ &\quad + \|\mathbb{1} - X\|_1 + 1 - \cos \langle Q, X \rangle) / (n + 1) \\ &\quad + 1 / (6.5 * \ln(N) - n) \end{aligned} \quad (2)$$

where  $N$  stands for the size of the dataset, and  $1 / (6.5 * \ln(N) - n)$  is a regularization term we found works well for preventing the path from being excessively long and meandering, and henceforth unrealistic, misleading, and confusing. The term adds a penalty to each node on the path and the penalty grows as the path evolves. A long path has a much larger  $g$  value than a short path with the regularization term being applied. As a result, our algorithm prefers shorter paths. We require this term because our  $g$  function evaluates the overall quality of the path instead of the accumulation of historical costs as A\* search does. It keeps no knowledge how long it has already traveled; therefore, a penalty for choosing too many nodes is necessary to control the path length.

The heuristic function  $h(x)$  is defined as the distance of  $X$  to  $Q$  plus the anticipated distance to the target node  $T$ :

$$h(X) = \|Q_{DCM} - X_{DCM}\|_2 + \|X_{DCM} - T_{DCM}\|_2 \quad (3)$$

Therefore,  $f(x)$  is

$$\begin{aligned} f(X) &= h(X) + g(X) \\ &= \|Q_{DCM} - X_{DCM}\|_2 + \|X_{DCM} - T_{DCM}\|_2 \\ &\quad + (g_n * n + \|Q_{DCM} - X_{DCM}\|_2 \\ &\quad + \|\mathbb{1} - X\|_1 + \cos \langle Q, X \rangle) / (n + 1) \\ &\quad + 1 / (6.5 * \ln(N) - n) \end{aligned} \quad (4)$$

With  $f(x)$  fully defined the search steps are similar to A\* search with several optimizations.

Fig. 6 shows some paths we generated for the three DCM enhancements and seven attributes. The stacked area charts

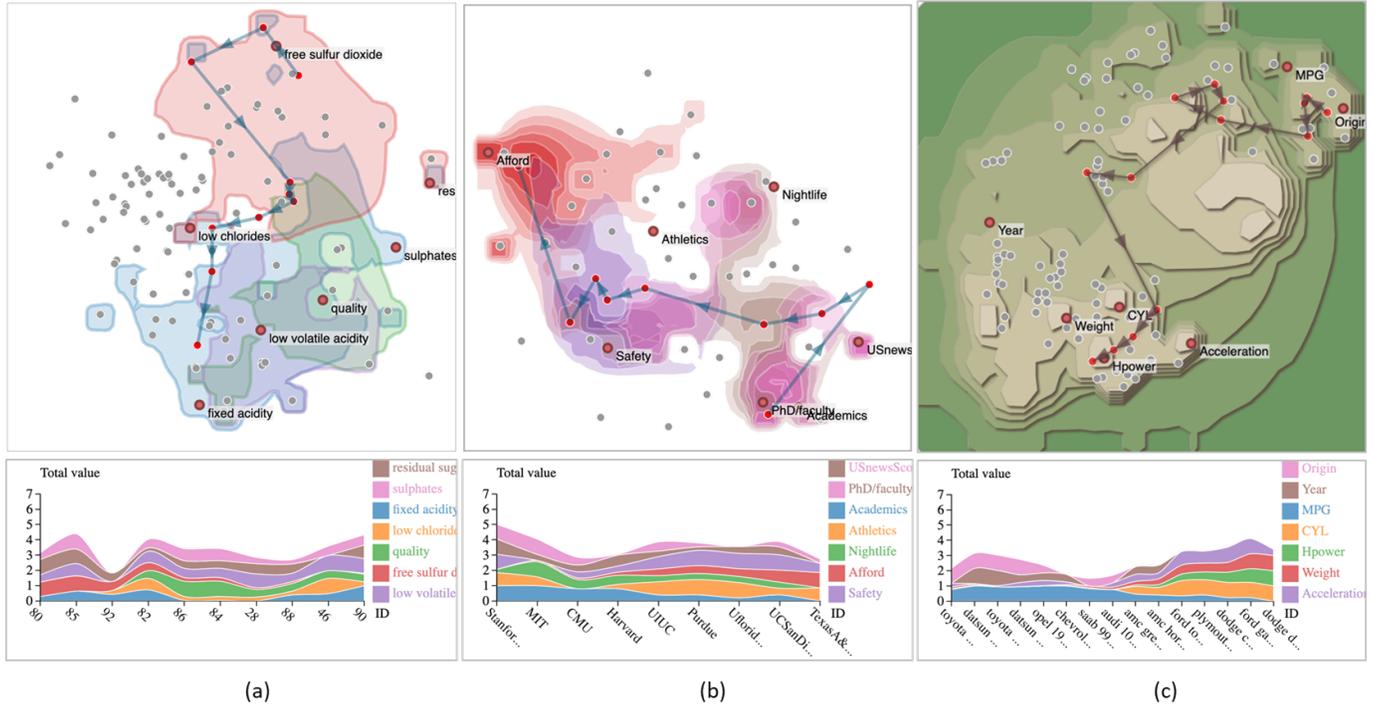


Fig. 6: Auto-generated Pareto paths for 7 attributes each. (a) The wine dataset visualized with the single iso-contour enhancement, (b) the university dataset visualized with the value-shaded topographic rendering, and (c) the car dataset with the topographic terrain rendering.

visualize the rich diversity of these benchmark exemplars. Note that for simplicity we only show the iso-contours and value-shading for a subset of the attributes in Fig. 6(a, b). The terrain in Fig. 6(c) is built based on all of them. The ability of taking a larger number of attributes into account is a strength of the terrain-based DCM.

---

#### Algorithm 1: Pareto based A\* search

---

```

Obtain the data matrix  $DM'$ ;
Initialize an open list  $[X_{source}]$  and a close list  $[\ ]$ ;
while open list is not empty do
   $Q \leftarrow$  element from open list with the smallest  $f(x)$ ;
  Remove  $Q$  from open list;
  Push  $Q$  to close list;
   $Q_c \leftarrow$  results from criterion optimization for
   $Q_{adjacent}$ ;
   $successors \leftarrow$  results from distance optimization for
   $Q_c$ ;
  for succ in successors do
    if succ is the target then
      | stop
    else
      |  $succ.g \leftarrow g(x)$ ;
      |  $succ.h \leftarrow h(x)$ ;
      |  $succ.f \leftarrow f(x)$ ;
    end
    if succ found in open list with smaller  $f$  or succ
    found in close list with smaller  $f$  then
      | skip
    else
      | push or update succ to open list
    end
  end
end
end

```

---

## 7 USER STUDY

We recruited 32 participants to test the four DCM designs. The subjects were divided into four groups and each tested one of the DCM variations with four datasets presented in random order: Pokémon, university, car and wine. We chose this between-subject design to prevent learning effects from occurring. Learning effects are plausible since all of our designs are derived from the DCM. An early pilot study in fact suggested that this might be the case.

Each dataset had a specific real-life application scenario that called for a quite well defined set of exemplars to be selected. The domain constraints, such as slider settings and the two terminal exemplars were held the same for each dataset to ensure users tested on the same conditions. Hence, the Pareto benchmark set was also constant for each dataset. The two terminal exemplars were spaced far apart in the layout and were very different. Thus we expected exemplars with good diversity in each set.

At the beginning of the user study, the participants were introduced to the DCM, one of the four graphical designs and the application scenarios for all four datasets. Following they were encouraged to learn about the two terminal exemplars and the attribute constraints from the UI. They could ask questions to make sure they fully understood the task. Once this introductory step was concluded, they were asked to find 7-10 good exemplars that represented a tradeoff between the two terminal exemplars as well as the ones selected thus far while keeping aware of the constraints.

The participants were recruited from different occupations, such as software engineer, company employee, hardware developer and PhD student. There were 18 males and 14 females; 2 participants were aged between 8-21, 14 between 22-25 and the rest between 26-29. Half of the participants had no visualization background while 2 had specific visualization expertise. Gender, age and visualization expertise were uniformly distributed within each group.

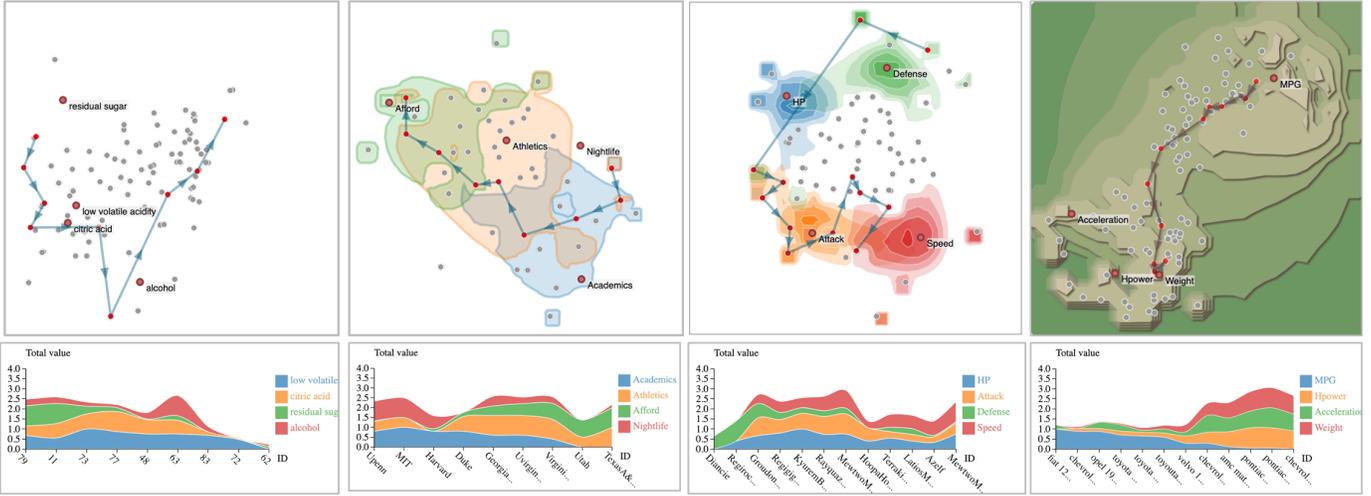


Fig. 7: The four scenarios for the user study, each rendered with one of the DCM graphical enhancements with the two terminal exemplars shown at the ends of the auto-computed Pareto path. (a) Party Host using the wine dataset, (b) College Counselor using the university dataset, (c) Pokémon Player using the Pokémon dataset, and (d) Rental Car Agency Manager using the car dataset.

### 7.1 Application Scenarios

**Pokémon Player:** In this scenario the participants were asked to slip into the role of a Pokémon player who was seeking to capture a lineup of complementary Pokémon characters. The Pokémon set consists of 65 Pokémon and six numerical attributes; the participants were told that the player thinks highly of four attributes, namely, Attack, Defense, HP and Speed. From the given DCM a participant would typically observe that no Pokémon could fit the constraints simultaneously and that compromise was unavoidable. While one Pokémon would perform well on Attack, the other was better on Defense with HP and Speed also playing a role. The participants were asked to help the player find further Pokémon characters that would make for a well-balanced lineup. In addition they were told that the player’s budget was limited to 12 characters including the two pre-specified extremes; 12 was the number of characters identified by our benchmark analysis (see Fig. 7(c)).

**College Counselor:** Here a college counselor was looking to compile a well balanced portfolio of colleges to contract with. We used a university data set of 45 schools and 15 parameters [47]. To keep things simple in conversations with students and parents, the counselor selects four criteria she feels play a dominant role: Academics, Athletics, Nightlife and Affordability. The participants were asked to predict which colleges the counselor might choose. They were given an upper limit of 9 colleges, the number chosen by the benchmark Pareto scheme (see Fig. 7(b)).

**Rental Car Agency Manager:** The participants were told of a fictitious rental car agency with a capacity of up to 12 cars and were asked to come up with a set of cars that would please most potential renters. There were 100 cars to choose from and the participants had the following four criteria at their disposal: Acceleration, Horsepower (HP), MPG and Weight. The DCM reveals that HP is inversely related to MPG leaving the two attributes on opposite ends on the DCM and making them mutually exclusive. Acceleration and Weight are positively correlated with HP and so end up mapped in a close DCM neighborhood. The Pareto path algorithm’s results are displayed in Fig. 7(d).

**Party Host:** In this last scenario the participants were to imagine hosting a large party and having to select a well-diversified selection of wines. They were told that the bar had the capacity to keep up to 9 variants of wines ready for consumption. The wine

data set has 100 variants of wine and 12 criteria for evaluation. The host picked 4 of these 12 criteria for the DCM which is plotted in Fig. 7(a) for the path returned by the Pareto optimization algorithm.

### 7.2 Results

Fig. 8 shows a sample of the exemplar sets selected by our participants, one set for each of the four DCM designs we studied. We evaluated each of these collected sets via this score function:

$$score = \frac{\sum_{i=1}^{n_p} \max_{0 < j \leq n_u} (dotproduct < e_p, e_{u(j)} >)}{n_p} \quad (5)$$

where  $n_p$  and  $n_u$  are the number of Pareto and user-chosen exemplars, respectively, for a given dataset and design. The score compares each Pareto exemplar with the closest exemplar chosen by the user. We obtain eight user scores per design and dataset, 128 scores in total. The strip plot in Fig. 9 shows these scores in detail. It is immediately obvious that the native DCM which acts as a baseline has a far lower median score than the three enhanced designs. It also appears that the iso-contour design performs lower than either of the two graphically more sophisticated designs. On the other hand, the performance is similar for these latter methods.

As mentioned, we chose a between-subject experiment where each group only tested one graphical design with four different datasets in random order. Considering Fig. 9 we can observe that there are wide score deviations across the four datasets, but less so within. It appears that our participants did exceedingly well with the university dataset. While the deviation among the other three datasets is less pronounced, a slight differentiation is still evident, with the car dataset performing worst. Nevertheless, the median performances of the four designs seems fairly consistent for each dataset which is encouraging.

The strip plots in Fig. 9 also reveal that the scores for each design and dataset are not normally distributed, calling for the use of a non-parametric statistical test. We first considered the Mann–Whitney U test taken over all four datasets, but this was not possible since the data sets varied about different median (as discussed above). We therefore applied the Aligned Rank Transform ANOVA (ART ANOVA) which aligns the data first

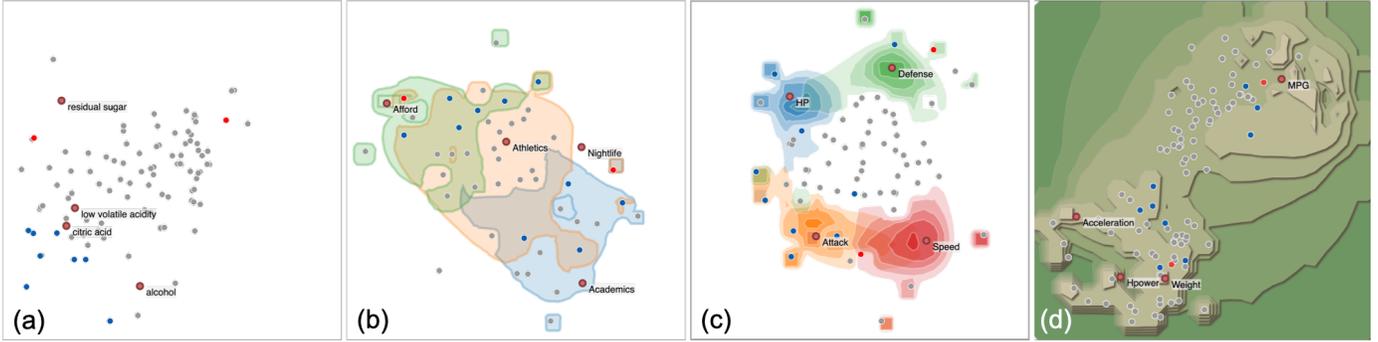


Fig. 8: 4/128 exemplar sets marked by 4/16 participants in our user study. In each figure the two red dots represent the two terminal exemplars while the blue dots are user selections. Each of these scenarios (a-b) corresponds to the respective scenario shown in Fig. 7.

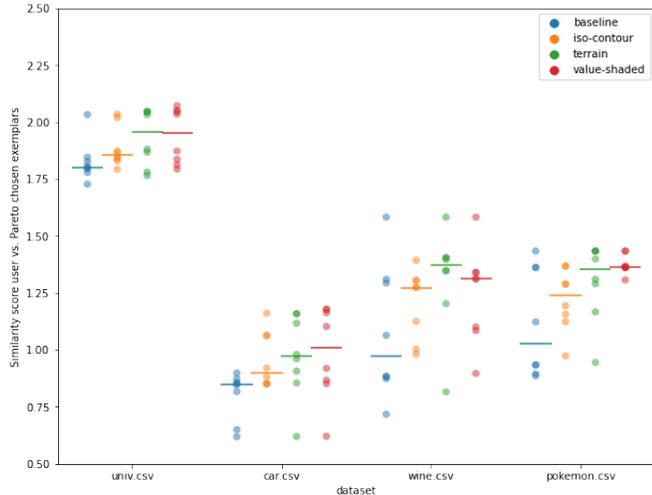


Fig. 9: Evaluation results. Each dataset has four strips with different colors that indicate the designs (see legend top right). Each bar in the strip represents the median score for that dataset and design.

before the ranks are calculated. The results are captured in Table 1. We observe a significant effect of the designs on evaluation score, and we also observe a significant effect of the datasets. There is, however, no interaction effect (des\*dat) between the datasets and the designs. All of these quantitative outcomes are also directly observable in the strip plots of Fig. 9, as discussed above.

To study the effects of the individual designs we conducted a post-hoc test and the results are shown in Table 2. We observe that the lower performance of the baseline design (the original DCM) with regards to all designs proposed in this paper is statistically significant ( $p \ll 0.05$ ). The difference among the three new designs, however, is more nuanced. While the terrain outperforms the iso-line design at  $p < 0.05$ , the significance of the value-shaded design is just below that level, at  $p = 0.058$ . This somewhat subtle difference in performance of the two designs with regards to the iso-line design is also evident by their statistically insignificant  $p = 0.8705$  when directly compared.

## 8 DISCUSSION

It is encouraging to see that any of the three graphical enhancements we proposed can provide users with a significantly better appreciation of the attribute-contextualized data visualization generated by the DCM than the original DCM. Furthermore, there is also statistical evidence that the topographic terrain map is superior to

	df	df.res	F	P
Designs	3	112	9.35	< 0.0001
Datasets	3	112	107.76	< 0.0001
Des*Dat	9	112	0.49	0.88

TABLE 1: The result of the two-way ART ANOVA test. The p-values show whether there is a statistically significant effect. We regard it statistically significant if  $p < 0.05$ .

design_pairwise	estimate	SE	df	t.ratio	p.value
baseline - iso-contour	-22.56	8.8	112	-2.57	0.0117
baseline - terrain	-40.81	8.8	112	-4.64	< 0.0001
baseline - value-shaded	-39.38	8.8	112	-4.48	< 0.0001
iso-contour - terrain	-18.25	8.8	112	-2.07	0.0403
iso-contour - value-shaded	-16.81	8.8	112	-1.91	0.0586
terrain - value-shaded	1.44	8.8	112	0.163	0.8705

TABLE 2: Post-hoc results of the 6 possible design pairings. We consider a difference statistically significant if the p-value  $< 0.05$ .

the colored iso-contour map which can be fairly easily derived from the DCM. Yet, there is no statistically significant difference between the topographic terrain map and the value-shaded topographic map. This is a positive outcome since the terrain map is more scalable than the value-shaded map and thus more general.

The iso-contour and value-shaded maps can get confusing when a large number of contour lines or colored areas end up crowding the space. By comparison, the terrain map is more scalable since it aggregates the various sets of iso-contour regions and creates the terrain from them. Higher elevations then denote regions where either one attribute highly dominates or where multiple attributes have favorable values. Looking at the proximity of attributes to these elevated regions can help users distinguish among these two cases. Data objects on a peak with one attribute sitting on top are likely dominated by that attribute and are more extreme in nature. Conversely, data objects on an elevated plateau or a saddle region with several attribute nodes in proximity are likely to be data objects that are strong in multiple attributes but not extreme in any of them. These are good candidates for exemplars. Additional attributes would then either give rise to individual peaks, if uncorrelated to any existing attribute, or else harden existing saddles or plateaus. This behavior makes the terrain preferable for larger numbers of attributes.

An inherent limitation of our evaluation is that our Pareto frontier-based algorithm, which generated the benchmark data, had access to much more comprehensive spatial information, namely the high-D data, than a human user who could observe the DCM

embedding of the data only. Despite this advantage we were still able to observe clear patterns in the user performance for different graphical enhancements, as described in the previous section. Making these determinations was the goal of this work.

Going forward, we believe there could be an opportunity to integrate the algorithmically generated Pareto frontier, or segments of it, into the DCM as initial suggestions which the user might then refine as desired. This has high potential for making the search for good exemplars more efficient and accurate, and at the same time still affords the user to make preferential choices, yet better informed. New research would be needed to determine how these suggestions are integrated into the display and to what extent, to avoid biasing the user to the algorithm's choices. It might be advisable to choose a denser set of suggested exemplars, possibly colored by their quality, as extracted from the algorithm.

It was interesting to see that the terrain display was significantly better than the iso-contour display but the value-shaded display was not. To boost the performance of the latter one might make use of textures such that overlapping regions could be better differentiated and visualized [58]. Yet another option could be to assign specific colors to certain multivariate value configurations [15].

Shortcomings of our method are similar to embedding systems in general: as the number of variables increases the mapping becomes increasingly inaccurate. This gets even worse when the attribute context needs to be preserved as well. Scalability to larger number of attributes can be achieved by clustering, followed by a multi-scale semantic zooming approach as described in [70]. There, a set of correlated attributes is replaced by the most dominant attribute, as determined by its interaction with other attributes, or by a specific hypernym attribute [41]. This in essence is similar to "chunking" which Miller [46] describes as the method used by humans to deal with a greater amount of information, such as grouping single letters into words or grouping single-digit numbers into blocks of multi-digit numbers. Prior dimension reduction or an interactive dimension selection interface to reduce the number of variables a manageable number can also be a viable option.

## 9 CONCLUSION

Our paper presented a novel visual interface not only for exemplar selection but also for multi-criteria decision making in general. It enables the decision maker to constrain criteria and then evaluate the existing trade-offs and the data objects that realize them. Traditional Pareto optimization methods typically work outside the user's control and are somewhat disconnected from a user's penchant to express personal preferences and intuition. Our approach puts the user into the loop by way of a dedicated visualization interface. It features a graphically enhanced map-like layout and a set of scented widgets which an analyst can use to express personal preferences and explore the available choices. Once expressed, these choices are then visualized as a path in the map display. The path can subsequently be utilized to populate a stacked area chart which conveys the trade-offs as smooth transitions. Our user study revealed that the graphical enhancements we designed help users in this task. In the future we hope to expand the user study further to gain more insight into the scalability of our method. Finally, we believe that these graphical enhancements might also be helpful to address similar tasks in other Attribute-Contextualized Data Visualizations, such as RadViz and Dust+Magnet. We plan to study this capacity in the future.

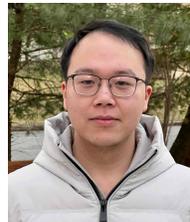
## ACKNOWLEDGMENTS

Support was provided by NSF grants IIS 1527200 and 1941613.

## REFERENCES

- [1] Auto MPG Dataset. <https://archive.ics.uci.edu/ml/datasets/auto+mpg/>. Accessed on 2021-07-07.
- [2] Radar Chart. [https://datavizcatalogue.com/methods/radar\\_chart.html/](https://datavizcatalogue.com/methods/radar_chart.html/). Accessed on 2021-07-04.
- [3] The Complete Pokemon Dataset. <https://www.kaggle.com/rounakbanik/pokemon>. Accessed on 2021-07-08.
- [4] H. Abbass, R. Sarker, and C. Newton. Pde: a pareto-frontier differential evolution approach for multi-objective optimization problems. In *Proc. Congress on Evolutionary Computation*, vol. 2, pp. 971–978 vol. 2, 2001.
- [5] V. Berezkin, G. Kamenev, and A. Lotov. Hybrid adaptive methods for approximating a nonconvex multidimensional pareto frontier. *Computational Mathematics and Mathematical Physics*, 46(11):1918–1931, 2006.
- [6] X. Blasco, J. Herrero, J. Sanchis, and M. Martínez. A new graphical visualization of n-dimensional pareto front for decision-making in multiobjective optimization. *Information Sciences*, 178(20):3908–3924, 2008.
- [7] P. Bourke. Conrec a contouring subroutine. [paulbourke.net/papers/conrec/](http://paulbourke.net/papers/conrec/). Accessed on 2021-07-10.
- [8] B. Broeksema, A. Telea, and T. Baudel. Visual analysis of multi-dimensional categorical data sets. In *Computer Graphics Forum*, vol. 32, pp. 158–169, 2013.
- [9] M. Chalmers. Using a landscape metaphor to represent a corpus of documents. In *European Conference on Spatial Information Theory*, pp. 377–390, 1993.
- [10] J. M. Chambers. *Graphical methods for data analysis*. CRC Press, 2018.
- [11] J. Chen and A. Sayed. Distributed pareto optimization via diffusion strategies. *IEEE Journal of Selected Topics in Signal Processing*, 7(2):205–220, 2013.
- [12] S. Chen, D. Amid, O. Shir, L. Limonad, D. Boaz, A. Anaby-Tavor, and T. Schreck. Self-organizing maps for multi-objective pareto frontiers. In *IEEE Pacific Visualization Symposium (PacificVis)*, pp. 153–160, 2013.
- [13] S. Cheng and K. Mueller. The data context map: Fusing data and attributes into a unified display. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):121–130, 2016.
- [14] S. Cheng, W. Xu, and K. Mueller. Radviz deluxe: An attribute-aware display for multivariate data. *Processes*, 5(4):75, 2017.
- [15] S. Cheng, W. Xu, and K. Mueller. Colormap nd: A data-driven approach and tool for mapping multivariate data to color. *IEEE Transactions on Visualization and Computer Graphics*, 25(2):1361–1377, 2018.
- [16] A. Cockburn. Revisiting 2D vs 3D implications on spatial memory. *Proc. Conference on Australasian User Interface*, 28:25–31, 2004.
- [17] A. Cockburn and B. McKenzie. An evaluation of cone trees. In *People and Computers XIV: BCS-HCI*, pp. 425–436. Springer Verlag, 2000.
- [18] E. Dimara, A. Bezerianos, and P. Dragicevic. Conceptual and methodological issues in evaluating multidimensional visualizations for decision support. *IEEE Transactions on Visualization and Computer Graphics*, 24(01):749–759, 2018.
- [19] F. Duchoň, A. Babinec, M. Kajan, P. Beňo, M. Florek, T. Fico, and L. Jurišica. Path planning with modified a star algorithm for a mobile robot. *Procedia Engineering*, 96:59–69, 2014.
- [20] R. Faust, D. Glickenstein, and C. Scheidegger. Dimreader: Axis lines that explain non-linear projections. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):481–490, 2018.
- [21] L. Ferariu and C. Cimpanu. Multiobjective hybrid evolutionary path planning with adaptive pareto ranking of variable-length chromosomes. In *2014 IEEE 12th international symposium on applied machine intelligence and informatics (SAMI)*, pp. 73–78. IEEE, 2014.
- [22] M. Friendly and D. Denis. The early origins and development of the scatterplot. *Journal of the History of the Behavioral Sciences*, 41(2):103–130, 2005.
- [23] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit. Lineup: Visual analysis of multi-attribute rankings. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2277–2286, 2013.
- [24] M. Gronemann and M. Jünger. Drawing clustered graphs as topographic maps. In *Proc. Graph Drawing*, pp. 426–438, 2013.
- [25] D. Harabor and A. Grastien. Online graph pruning for pathfinding on grid maps. In *Proc. AAAI Conference on Artificial Intelligence*, vol. 25, 2011.
- [26] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

- [27] J. A. Hartigan. Printer graphics for clustering. *Journal of Statistical Computation and Simulation*, 4(3):187–213, 1975.
- [28] K. Hinum, S. Miksch, W. Aigner, S. Ohmann, C. Popow, M. Pohl, and M. Rester. Gravi++: Interactive information visualization to explore highly structured temporal data. *J. Univers. Comput. Sci.*, 11(11):1792–1805, 2005.
- [29] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley. Dna visual and analytic data mining. In *Proc. IEEE Visualization*, pp. 437–441, 1997.
- [30] M. Höggräfer, M. Heitzler, and H.-J. Schulz. The state of the art in map-like visualization. In *Computer Graphics Forum*, vol. 39, pp. 647–674, 2020.
- [31] C.-L. Hwang and A. S. M. Masud. *Multiple objective decision making—methods and applications: a state-of-the-art survey*, vol. 164. Springer Science and Business Media, 2012.
- [32] S. Ingram, T. Munzner, and M. Olano. Glimmer: Multilevel mds on the gpu. *IEEE Transactions on Visualization and Computer Graphics*, 15(2):249–261, 2008.
- [33] A. Inselberg and B. Dimsdale. Parallel coordinates for visualizing multi-dimensional geometry. In *Computer Graphics*, pp. 25–44, 1987.
- [34] E. Kandogan. Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions. In *Proc. IEEE Information Visualization Symposium*, vol. 650, p. 22, 2000.
- [35] G. A. Kiker, T. S. Bridges, A. Varghese, T. P. Seager, and I. Linkov. Application of multicriteria decision analysis in environmental decision making. *Integrated Environmental Assessment and Management: An International Journal*, 1(2):95–108, 2005.
- [36] M. M. Köksalan, J. Wallenius, and S. Zionts. *Multiple Criteria Decision Making: From Early History to the 21st Century*. World Scientific, 2011.
- [37] M. Kraus, K. Angerbauer, J. Buchmüller, D. Schweitzer, D. A. Keim, M. Sedlmair, and J. Fuchs. Assessing 2D and 3D heatmaps for comparative analysis: An empirical study. In *Proc. ACM CHI*, pp. 1–14, 2020.
- [38] J. Kruskal and M. Wish. *Multidimensional Scaling*. Sage Publications, 1978.
- [39] A. Lavin. A pareto front-based multiobjective path planning algorithm. *arXiv preprint arXiv:1505.05947*, 2015.
- [40] A. V. Lotov and K. Miettinen. *Visualizing the Pareto Frontier*, pp. 213–243. Springer Berlin Heidelberg, 2008.
- [41] S. Mahmood and K. Mueller. Taxonomizer: Interactive construction of fully labeled hierarchical groupings from attributes of multivariate data. *IEEE Transactions on Visualization and Computer Graphics*, 26(9):2875–2890, 2019.
- [42] K. Marriott, J. Chen, M. Hlawatsch, T. Itoh, M. A. Nacenta, G. Reina, and W. Stuerzlinger. Immersive analytics: Time to reconsider the value of 3d for information visualisation. In *Immersive Analytics*, pp. 25–55. Springer, 2018.
- [43] L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
- [44] M. Meyer, A. Barr, H. Lee, and M. Desbrun. Generalized barycentric coordinates on irregular polygons. *Journal of Graphics Tools*, 7(1):13–22, 2002.
- [45] K. Miettinen. *Nonlinear Multiobjective Optimization*, vol. 12. Springer Science and Business Media, 2012.
- [46] G. A. Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63(2):81, 1956.
- [47] E. J. Nam and K. Mueller. Tripadvisor<sup>ND</sup>: A tourism-inspired high-dimensional space exploration framework with overview and detail. *IEEE Trans on Visualization and Computer Graphics*, 19(2):291–305, 2012.
- [48] A. Nash and S. Koenig. Any-angle path planning. *AI Magazine*, 34(4):85–107, 2013.
- [49] M. Nasrolahzadeh, A. Ibrahim, S. Rahnamayan, and J. Haddadnia. Pareto-radviz: A novel visualization scheme for many-objective optimization. In *IEEE Intern. Conference on Systems, Man, and Cybernetics (SMC)*, pp. 3868–3873, 2020.
- [50] P. Oesterling, G. Scheuermann, S. Teresniak, G. Heyer, S. Koch, T. Ertl, and G. H. Weber. Two-stage framework for a topology-based projection and visualization of classified document collections. In *IEEE Symp. on Visual Analytics Science and Technology (VAST)*, pp. 91–98, 2010.
- [51] K. A. Olsen, R. R. Korfhage, K. M. Sochats, M. B. Spring, and J. G. Williams. Visualization of a document collection: The vibe system. *Information Processing & Management*, 29(1):69–81, 1993.
- [52] I. Ozsahin, D. U. Ozsahin, and B. Uzun. *Applications of Multi-Criteria Decision-Making Theories in Healthcare and Biomedical Engineering*. Elsevier Science and Technology, 2021.
- [53] L. d. C. Pagliosa and A. Telea. Radviz++: Improvements on radial-based visualizations. In *Informatics*, vol. 6, p. 16, 2019.
- [54] S. Pajer, M. Streit, T. Torsney-Weir, F. Spechtenhauser, T. Möller, and H. Piringer. Weighlifter: Visual weight space exploration for multi-criteria decision making. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):611–620, 2017.
- [55] G. Robertson, M. Czerwinski, K. Larson, D. C. Robbins, D. Thiel, and M. Van Dantzich. Data mountain: using spatial memory for document management. In *Proc. ACM UIST*, pp. 153–162, 1998.
- [56] G. Robertson, J. Mackinlay, and S. Card. Cone trees: animated 3d visualizations of hierarchical information. In *Proc. ACM CHI*, pp. 189–194, 1991.
- [57] Y. Sawaragi, H. Nakayama, and T. Tanino. *Theory of Multiobjective Optimization*. Elsevier, 1985.
- [58] H. Shenias and V. Interrante. Compositing color with texture for multivariate visualization. In *GRAPHITE*, pp. 443–446, 2005.
- [59] J. Soo Yi, R. Melton, J. Stasko, and J. A. Jacko. Dust & magnet: multivariate information visualization using a magnet metaphor. *Information Visualization*, 4(4):239–256, 2005.
- [60] M. Tory, D. Sprague, F. Wu, W. Y. So, and T. Munzner. Spatialization design: Comparing points and landscapes. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1262–1269, 2007.
- [61] M. Tory, C. Swindells, and R. Dreezer. Comparing dot and landscape spatializations for visual memory differences. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1033–1040, 2009.
- [62] L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.
- [63] T. van Dijk, A. van Goethem, J.-H. Huanert, W. Meulemans, and B. Speckmann. Map schematization with circular arcs. In *Geographic Information Science*, pp. 1–17, 2014.
- [64] P. Van Kerm. Adaptive kernel density estimation. *The Stata Journal*, 3(2):148–156, 2003.
- [65] D. A. Van Veldhuizen, G. B. Lamont, et al. Evolutionary computation and convergence to a pareto front. In *Genetic Programming Conference, Late Breaking Papers*, pp. 221–228, 1998.
- [66] M. Velasquez and P. Hester. An analysis of multi-criteria decision making methods. *Int'l Journal of Operations Research*, 10(2):56–66, 2013.
- [67] L. Wilkinson and M. Friendly. The history of the cluster heat map. *The American Statistician*, 63(2):179–184, 2009.
- [68] W. Willett, J. Heer, and M. Agrawala. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1129–1136, 2007.
- [69] J. Wise, J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow. Visualizing the non-visual: Spatial analysis and interaction with information from text documents. In *Proc. IEEE Visualization*, pp. 51–58, 1995.
- [70] Z. Zhang, K. T. McDonnell, E. Zadok, and K. Mueller. Visual correlation analysis of numerical and categorical data on the correlation map. *IEEE Trans. on Visualization and Computer Graphics*, 21(2):289–303, 2014.



**Xinyu Zhang** received his bachelor's degree from Shandong University, Taishan College. Now he is a PhD student at Stony Brook University advised by Prof. Klaus Mueller. His current research field is multivariate data analysis.



**Shenghui Cheng** is a Westlake fellow at Westlake University. He received a PhD in computer science from Stony Brook University and held research positions in the Institute of Medical Research of the University of Leipzig, Germany, Brookhaven National Lab and Harvard Medical School. He founded the Dagoo data visualization platform and served as a mentor of the Stanford AI Global Executive Leadership Program as well as several other positions of that nature.



**Klaus Mueller** received the PhD degree in computer science from The Ohio State University. He is currently a professor of computer science at Stony Brook University and a senior scientist at Brookhaven National Lab. His current research interests include visualization, visual analytics, data science, and medical imaging. He has authored more than 200 papers which were cited more than 11,000 times. For more information, see <http://www.cs.stonybrook.edu/~mueller>.