IEEE TRANSACTIONS ON
VISUALIZATION AND
COMPUTER GRAPHICS

Scatterplots: Tasks, Data, and Designs

Alper Sarikaya, *Student Member, IEEE* and Michael Gleicher, *Member, IEEE*

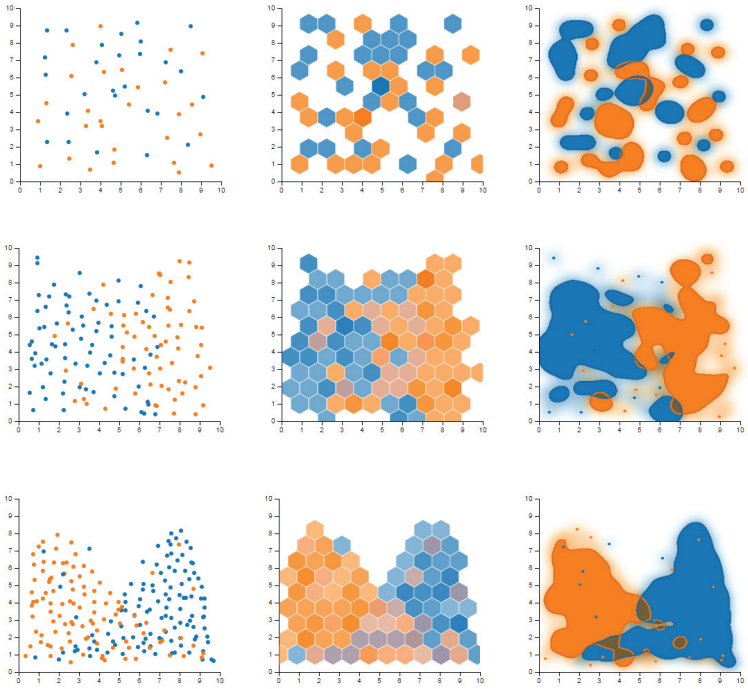


Fig. 1. Scatterplot designs (shown in columns) have varying levels of support for viewer tasks based on the data characteristics (rows). Here, we compare a traditional scatterplot (left column) to a hexagonal binning implementation [11] (middle) to a Splatterplot [49] (right) for three representative datasets. The appropriateness of a scatterplot design is based on the characteristics of the data and the design's support of the viewer's task (such as identifying outliers or comparing distributions). For random distributions with few points (top row), the traditional scatterplot (left) describes the data plainly. With increasing numbers of points (middle row), aggregation representations such as binning (center) communicate spatial density. With overlapping distributions (bottom row), density-based representations communicate overlap and can also show outliers (right), which disappear in the binned representation (middle).

Abstract—Traditional scatterplots fail to scale as the complexity and amount of data increases. In response, there exist many design options that modify or expand the traditional scatterplot design to meet these larger scales. This breadth of design options creates challenges for designers and practitioners who must select appropriate designs for particular analysis goals. In this paper, we help designers in making design choices for scatterplot visualizations. We survey the literature to catalog scatterplot-specific analysis tasks. We look at how data characteristics influence design decisions. We then survey scatterplot-like designs to understand the range of design options. Building upon these three organizations, we connect data characteristics, analysis tasks, and design choices in order to generate challenges, open questions, and example best practices for the effective design of scatterplots.

Index Terms—Scatterplots, task taxonomies, study of designs

1 Introduction

Scatterplots are a very common type of visualization. Their flexibility has led to their use in a variety of exploratory and presentation contexts. The traditional scatterplot represents each object in a dataset with a point (or other mark), positioned on two continuous, orthogonal dimensions. As data grows in scale and complexity, the traditional scatterplot design rapidly becomes ineffective. As a result, many other scatterplot designs have been proposed. While these designs may address scale, they are often specific to data characteristics and tasks. Designers have little guidance in how to select among design choices. Our goal is to help designers select scatterplot designs that are appropriate to their scenarios by identifying the factors that affect the appropriateness of scatterplot designs.

In this work, we describe how to consider analysis scenarios in terms of their task and data characteristics in order to determine which scatterplot designs are appropriate. We generate a framework by collecting and abstracting use cases of scatterplots in the literature. For *tasks,* we collect model tasks that are performed with scatterplots, creating an abstraction that helps us to understand the task space of scatterplots. We also identify a number of design-relevant *data characteristics,* such as the number of objects. To identify the space of potential *designs,* we survey scatterplot designs to organize and cluster similar design

- *Alper Sarikaya is with the University of Wisconsin—Madison. E-mail: sarikaya@cs.wisc.edu.*
- *Michael Gleicher is with the University of Wisconsin—Madison. E-mail: gleicher@cs.wisc.edu.*

Manuscript received 31 Mar. 2017; accepted 1 Aug. 2017.
Date of publication 28 Aug. 2017; date of current version 1 Oct. 2017.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2017.2744184

decisions together. We use tasks and data characteristics to reason about the applicability of these designs. Our framework, therefore, provides a process for designers to select scatterplot designs appropriate to their scenario by first identifying relevant task and data characteristics. Additionally, the framework highlights areas in the design space for further exploration, and where multiple solutions exist for similar, abstract problems.

The framework that we construct in this paper uses analysis task and data characteristics to identify the scenarios in which a design is appropriate, much like the methodology championed in Munzner's text [51]. Through the paper, we will summarize a short history of the scatterplot and related designs (§2.1), orient our framework relative to existing visualization taxonomies (§2.2), frame and identify the relevant factors that affect scatterplot design (tasks [§3] and data characteristics [§4]), survey the space of designs (§5), and explore how the framework can be used to determine the appropriateness of scatterplot designs (§6–7).

2 Background

2.1 Scatterplots

The scatterplot was designed to emphasize the spatial distribution of data plotted in two-dimensions. While the scatterplot itself has had a long history (see Friendly and Dinis [29]), its relative simplicity and flexibility enables the scatterplot as an ideal sandbox for early information visualization and perceptual psychology research. In particular, Cleveland [16] notes three factors that may affect the design decisions that are made by the designer of a scatterplot: (1) marks or points are designed with preattentive features in mind, (2) scatterplots are designed with the detection of individual objects in mind, and also (3) are designed such that the distances between objects represent a notion of similarity. With different sets of guiding factors, many different variations around the core scatterplot design have been developed—many trying to squeeze more fidelity from the traditional mark-per-object, two-dimensional scatterplot design. These designs are typically at odds with factor (2) above, prioritizing aggregate judgments over object-centric affordances. In this section, we highlight the background of challenges in adapting scatterplots to different analysis scenarios. While some of these strategies utilize multiple scatterplots, linked components, or glyphs as marks, we only consider the use of a single, two-dimensional, scatterplot with mono-variate marks outside of this section for the purposes of concision.

Dealing with too much data—Scatterplots work very well for a variety of analyses—until the amount of data overwhelms the traditional design of assigning a mark to every datum in a dataset. *Overdraw* is a common concern for scatterplots, defining the scenario where marks overlap one another and mask marks drawn under them. Cui *et al.* [22] notes that the drawing order can have serious ramifications of emphasizing inaccurate judgments of distribution. Fekete and Plaisant [27] highlight technical issues in displaying millions of items, where overdraw is a prime concern.

Reducing the data is one approach to address the challenge of too much data. Strategies include reducing the data before mapping to a visual representation, simplifying the visual representation itself, or modifying the space of the plot. In the first case, stochastic or stratified subsampling of the data [4, 14] is an example of reducing the number of data for display. Binning data [17] by collecting counts within small localized regions and visualizing area-aggregated, relative counts is another strategy in this same vein. Strategies to simplify the visual representation, such as continuous density estimation used by contour plots [19], landscape maps [68], and Splatterplots [49], aggregate marks by their position, highlighting clusters and distributions of marks.

Modifying the space of the plot can also emphasize hidden structures. Generalized scatterplots [37] and related work (e.g., continuous scatterplots [3]) take advantage of open space in a plot by performing a subspace warp to take advantage of unused regions of the graph, while combining the strengths of density estimation. In addition to these techniques, organizations of the strategies have been proposed, most notably Ellis and Dix's work [25] on clutter reduction where many strategies are directly applicable to scatterplot data. In this paper, we provide organization of the factors specific to scatterplots that can assist in selecting these types of design elements and techniques from the possible space of all designs.

Dealing with high-dimensional data— Scatterplots have enjoyed continued use in the visualization of high-dimensional data. Brehmer *et al.* [9] outline some of the analysis scenarios covered by scatterplots and related visualizations. Using scatterplots, the three common strategies are to select a subset of two dimensions, reduce the dimensionality to two dimensions using a dimensionality reduction technique, or showing all dimensions in a pairwise fashion. In the first case, simply showing a subset of two dimensions reduces to the typical scatterplot use case, though the distance between marks only communicates similarity in a reduced subspace.

Commonly, dimensionality reduction methods use scatterplots to visualize their results. Techniques may project points using such a method to cluster similar objects together, such as the work by Lehmann *et al.* [42] and Yuan and co-authors [75]. Some other scatterplot-related designs bridge the gap back to feed input back to dimensionality reduction techniques, such as Dis-Function [10] and InterAxis [38] by using direct manipulation to drive and update object clustering and projection functions.

SPLOMs [11] are a popular choice for visualizing pairwise dimensional information, highlighting correlations between pairs of dimensions. However, the paradigm does not scale well to high numbers of dimensions. In response, scagnostics [73] provide metrics for identifying interesting correlations and patterns in two-dimensional data, including features described as shape, trend, and coherence. These measures can be used to find interesting combinations of dimensions to visualize, as shown in both Bertini *et al.* [5] and Tatu *et al.* [66], and can be used to help guide interaction, as shown by Dang *et al.* [23]. To support increasing complexity in high-dimensional data, there have been variations on the SPLOM and scagnostic themes, including the use of radial graphs [36] to show all dimensions in a two-dimensional plane. Yates *et al.* [74] takes an additional step of abstracting the "shape" of pairwise correlation in individual scatterplots within a SPLOM, highlighting trends of correlation. Clearly, the support of dimensionally-reduced data is an important analysis case for scatterplots, but how to select between these possible strategies for scatterplot design is unclear, especially with the increased scale and complexity of datasets.

Designing for cognition and perception—There are yet other challenges faced by work that tackle how to design for data complexity in scatterplots. Central to many of these techniques is preserving the meaning of distance between objects as an indicator of similarity. In geography, the "first law of cartography" that states that objects closer in distance tend to be more similar [46], which has been adapted and codified to point spatializations (a.k.a, scatterplots) by Montello *et al.* [50]. In particular, work has concentrated on the aspect ratio of the plot area, which can affect judgments of distance between objects [17, 33, 65], as well as global judgments of correlation [43, 52]. Scatterplots have also been a canonical player in testing perceptual issues of different visual encodings, including probing just-noticeable differences in point lightness [45], point size [44], comprehension of group statistics between point classes [30], and the judgment of linear correlation [43]. Though the scope of these works concentrate on specific design decisions, combining these strategies can help to derive effective scatterplot design.

2.2 Typologies and Taxonomies

Typologies and taxonomies use abstraction to extract similarities and differences between concepts without unnecessary dependence on the particulars of individual implementations. A primary consideration common in many information visualization taxonomies is task, abstracting how a viewer interacts with and obtains information from a visualization (see Munzner [51] for a high-level overview). Task is typically viewed on a continuum from high- to low-level [54]: a high-level task comprises an analysis goal [8, 57], while a low-level task captures the exact information that viewers pull out of a visual representation [12, 32] or describes bite-sized analyses [2]. Another

consideration may be understanding how factors and characteristics of the data can have ramifications on the visualization, as discussed by both Mackinlay [47] and Ellis and Dix [25]. These taxonomies, along with many others, help to standardize the lexicon and assist in the incremental progress toward tailoring effective design for a given analysis goal.

These taxonomies discuss visualizations in a general case, making it difficult to apply these organizations to influence the designs of specific visualizations in practice—though exceptions exist: Sedlmair *et al.*'s taxonomy for dimensionality reduction [59], Sedlmair *et al.*'s taxonomy of cluster separation factors [60], and Lee *et al.*'s taxonomy for graph data [40]. By focusing on the single scatterplot case, the goal in this paper is to create a framework with an impact statement similar to the design space goal set out in Schulz *et al.* [57]. A framework should consolidate many similar but disparately presented research under a single lens to drive the framework forward, by explicitly examining the trade-offs between different strategies. By organizing research in this way, such a framework would concretize (or, in the words of Schulz *et al.*, externalize) implicit design decisions to explicitly organize how designs work—helping to teach practitioners and researchers, clarifying design requirements, and making good abstractions that have practical value. This also has the advantage of identifying open areas for future research by identifying voids in the design space—as an example, it may become clear through the organization that a strategy does not exist for a particular set of factors. Therefore, our goal throughout this work is to create a framework specific for scatterplot-like designs, helping both practitioners and tool-builders to choose the correct design, given both the analysis goal and the characteristics of their data.

3 Scatterplot Tasks

While many task taxonomies have been constructed for general information visualization or even for specific data types (e.g., graphs [40]), we are not aware of such a task analysis specific for scatterplots. With a coverage of the space of the analysis tasks that concern scatterplots, a task list can allow for discrimination between designs—helping to identify why one design may work better in a particular analysis scenario over another. We seek to identify tasks that form the building blocks for all analysis done with scatterplots, covering both low-level and high-level tasks. Such a list should be data domain-agnostic, which would allow for creating actionable abstractions of specialized scatterplot-like designs for specific data domains in similar analysis scenarios.

To formulate the seeds for this task list, we collected model tasks from a variety of sources in the data visualization literature, including papers performing empirical evaluation [52, 68], picking "good" views of correlation and clustering [43, 45, 61, 66], design studies of analyst scenarios [9, 59], technique papers [4, 21, 69], and even position papers [20]. The list of 23 collected model tasks, their source, and common categories are detailed in the supplemental material. To abstract these model tasks, we asked four data visualization researchers (two faculty, two senior doctoral students; 5–10 years of experience) to perform a card-sort and group tasks together based on their similarity (see Spencer for an introduction [62]). This card-sort strategy has precedent in the community—see Roth's application for deriving a set of cartographical interaction intents [55]. We asked them to use an open card-sort (no predefined categories titles, nor prescribed number of categories), and arrived with several categorizations of tasks. With minor disagreement, exploratory and cluster analysis generated consensus groups consisted of tasks that we then labeled *open-ended browsing and exploring*, *cluster rationalization*, *density judgments*, *dimension rationalization*, *multi-scatterplot tasks*, and *trend analysis*. Due to our concentration on single plot designs, we discarded the *multi-scatterplot tasks* category.

With these seed categories, we refined these categories post-hoc to generate a complete picture of the space (Table 1). We refocused the *trend analysis* category from the card-sort to *explore neighborhood* (#5), which captures obtaining aggregate statistics about a group [16, 30, 32], or identifying the similarities and differences among objects in a spatial region [68]. To expand the *browsing and exploring* category into representative tasks, we use Casner's taxonomy [12] to separate directed and undirected search, yielding the two tasks *search for known motif* (#6) and *explore data* (#7).

	#	Task	Description
object-centric	1	Identify object	Identify the referent from the representation
	2	Locate object	Find a particular object in its new spatialization
	3	Verify object	Reconcile attribute of an object with its spatialization (or other encoding)
	4	Object comparison	Do objects have similar attributes? Are these objects similar in some way?
browsing	5	Explore neighborhood	Explore the properties of objects in a neighborhood
	6	Search for known motif	Find a particular known pattern (cluster, correlation)
	7	Explore data	Look for things that look unusual, global trends
aggregate-level	8	Characterize distribution	Do objects cluster? Part of a manifold? Range of values?
	9	Identify anomalies	Find objects that do not match the 'modal' distribution
	10	Identify correlation	Determine level of correlation
	11	Numerosity comparison	Compare the numerosity/density in different regions of the graph
	12	Understand distances	Understanding a given spatialization (e.g. relative distances)

Table 1. Our list of abstracted analysis tasks that are performed with scatterplots: model tasks gathered from the literature, categorized with a card sort, and refined through reconcilation with visualization taxonomies.

Judgments of distribution are another common task—while many papers concentrate on finding clusters [9, 61, 66], identifying other distributions such as manifolds are also important in many analysis scenarios [45, 59]—giving rise to the *characterize distribution* task (#8). From the *browsing* category, we also explicitly partitioned the task of *identifying anomalies* (distributional-specific outliers, #9) due to the common trade-off of scatterplot designs utilizing aggregation [26]. *Identifying correlation* (#10) between the two dimensions in a scatterplot is a canonical task with scatterplots [16, 29], which has prompted empirical studies of how correlation is identified in scatterplots [43, 52]. We adapt the derived *density judgment* category to *numerosity comparison* (#11), which captures tasks that coarsely compare the numbers of objects embedded in spatial regions within the scatterplot. The last task in our list is *understand distances* (#12), capturing elements of the derived *dimension rationalization* category, to capture tasks of using and judging distances as a metric space against an object-embedded subspace [50, 59].

Many high-level tasks have been captured through this refinement process—dealing with sets of objects and understanding trends, distributions, and numerosity. We augment the derived tasks from the card-sort also to capture single-cardinality, object-centric tasks, such as look-up and *identifying* an object's spatialization (#1), searching for and *locating* an object in a scatterplot (#2), and *verifying* an object's spatialization within the plot (#3). As a pair to exploring the neighborhood (#5), *object comparison* (#4) involves comparing the visually-mapped (and non-mapped) attributes of a pair of objects to determine the relationship or similarity between the two data items [16]. These operations represent low-level operations in the visualization literature, stemming from Casner's analysis taxonomy [12] and repeated in others [8, 51, 57].

This process results in twelve abstract tasks (Table 1) that we use to help frame our discussion throughout this paper. This collection of tasks is the first derived collection of tasks that are specific to scatterplots, and abstracts the range of tasks from a variety of scatterplot designs. A complex analysis task performed within an analysis scenario, such as correlation discovery in high-dimensional data, may involve several of these tasks, used as building blocks, to achieve an analysis goal, similar

Data Attribute	Possible Values	Relevant Work
Class label	No class label, 2-4 classes, 5+ classes	[24, 31, 61]
Num. of points	Small (<10), medium (10–100), large (100–1000), very large (>1000)	[21,30,37,49,68]
Num. of dimensions	Two continuous, two derived, or >2 dimensions	[6, 13, 59]
Spatial nature	Dimensions do/do not map to spatial position	[46, 50]
Data distribution	Random, linear correlation, overlap, manifolds, clusters	[5, 23, 43, 52, 59, 61, 66, 73]

Table 2. The data attributes considered in our work, with work inspiring these distinctions. The number of points are quantized into bins based on their overdraw effect on design decisions—numbers given are relevant for the 400×400 plot and 6×6 mark sizes shown in this paper [70].

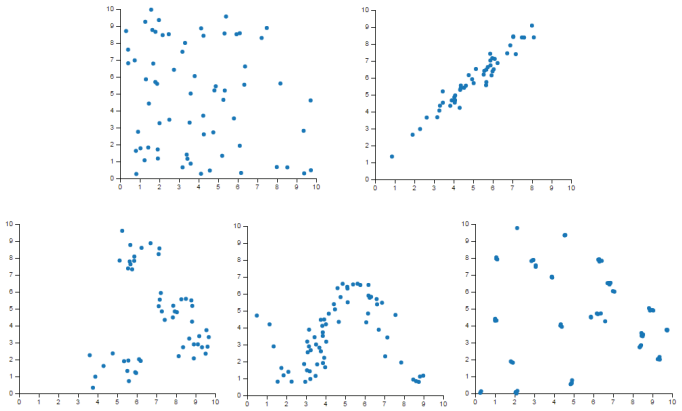


Fig. 2. Sample distributions captured with the five types of data distributions considered: randomly distributed, linear correlation, clustering, manifold (matching a discernable function), and overlapping points.

to the task construction presented in other task taxonomies [8, 57]. As an example, consider one of the model tasks collected: "match clusters and classes" (from Brehmer *et al.* [9]). This analysis goal can be composed of tasks #6, #5, and #4: search for known motif (find clusters), explore neighborhood (inspect objects within the cluster), and object comparison (inspect class membership of objects).

The curation of this list allows us to focus on supporting these tasks downstream in gauging and evaluating the task performance on scatterplots, based on data characteristics and design strategies. The task list helps to cover the range of tasks done with scatterplots over a wide range of analysis scenarios, but it is not able to capture how data characteristics may make some designs intractable. Nonetheless, we can use these tasks as a factor to distinguish the strengths and weaknesses of scatterplot-like designs, and we can provide better coverage of analysis scenarios when paired with data characteristics.

4 Data Characteristics

Many characteristics of the data (such as data size and distribution) may influence the design of an appropriate scatterplot. Similar to capturing the tasks of scatterplots, collecting, abstracting, and connecting relevant data characteristics will allow a more complete characterization of task effectiveness in the space of scatterplot designs. Here, we survey the challenges in particular sets of data characteristics, and discuss designs to support these characteristics in the design decisions (§5) and linking (§6) sections. The data attributes that we consider in this work are summarized in Table 2, along with reference articles.

There has been precedence of capturing relevant data characteristics in scatterplots, both explicitly and implicitly. Implicit representations develop responsive designs to varying data characteristics, such as encodings that scale to support increased numbers of points (see Sedlmair *et al.* [60] for relevant factors in cluster separation). Explicit representations use quantitative metrics to capture different features of data characteristics. For example, Wilkinson *et al*'s [73] re-introduction of scagnostics to the information visualization community allows for metric calculation of very particular distribution characteristics. Capturing these relevant characteristics can help to quickly whittle down the space of applicable techniques when considering a large combination of dimensions or a large space of scatterplot-like designs.

A common data characteristic that prompts design consideration is an increased number of objects to represent. A critical threshold in understandability is reached when the number of objects to visualize approaches the limit of available screen-space to show individual points. Very clearly, the *number of points* to consider in a scatterplot will affect the appropriateness of a design, dependent on the screen-space available for the plot (see Urribarri and Castro [70] for a discussion)—though the number of points may not affect some analysis tasks [30]. We quantize this factor into bins, where the bins prompt different design strategies to handle issues of data scale, such as the issue of overdraw. The data scale for the bins are dependent on the mark size and the plot size [70]—for example, larger plots with have higher thresholds for "large" numbers of points. Viewers can pick out individual marks and their referents at *small* numbers, while it is more difficult to pick out individual points at a *medium* scale. A *large* number of marks starts to exhibit problems of overdraw, while a *very large* number of points can only be displayed in aggregate. At larger data scales, designs tend to make use of aggregation to handle the data scale (§5).

Related to the number of objects, multiple data series are often shown in the same plot to compare distributions between and among groups [61]. A *class label* identifies different data series by discriminating points by shape or color. These labels can allow tasks to be performed on series in aggregate, which may or may not cause issues of distraction when performing tasks on individual series [24, 30, 31]. We discuss some relevant designs in the discussion (§7), though we do not consider multi-variate encodings (such as glyphs) in this paper.

While we concentrate on two-dimensional scatterplots in this paper, the *number of dimensions* of the objects under consideration is also an important consideration to make. We make a distinction between visualizing objects with two continuous dimensions [6], two derived dimensions (from a process such as principle-components analysis) [59], and visualizing a subset of dimensions from objects with more than two continuous dimensions (such as considering a high-dimensional system) [13], as these scenarios effect design choices. Depending on the dimensionality considered, some tasks such as understanding correlation and distribution of the data may need additional design scaffolding, requiring viewer interaction to understand correlation throughout the dataset. As another example, clustering takes on different meanings depending on whether the data is projected from some high-dimensional space or being positioned based on two attributes (absolute object similarity vs. subspace). Similarly, we can also consider dimensions that *do* and *do not map to spatial position* [50]; this distinction can affect how viewers interpret distance as object similarity in the plot.

Finally, the expected *distribution of the data* can affect the performance of various tasks—points can cluster, form distinct correlations, or can even stack. Scagnostics [73] provides a list of nine types of relationships between two continuous variables. We seek a more focused list of relationship categories that designs may target. These five categories are not necessarily exclusive to one another, but serve to separate how task appropriateness may be affected by the distribution of the data. Data that groups into **clusters** (*clumpy* in scagnostics) is an area of interest in the literature [61], as identifying why clusters occur can be an analysis task. The distribution of the data can also group into semantically-meaningful shapes such as **manifolds** (*coherence* in scagnostics), which can be relevant for other analyst tasks [9, 59]. The potential for overdraw increases if the distribution of the data involves points that have very **similar dimensional values**, and scagnostics captures this sentiment with *clumpiness*. Data that contains a **linear correlation** (*trend* in scagnostics) are critical to communicate

effectively, and much research in both the statistics and information visualization communities has focused on good design decisions to emphasize potential correlation, including picking the ideal aspect ratio for the plot [28, 53] and adding visual embellishments such as a trendline. Lastly, data that appears randomly distributed (where a discernable trend is hard to determine) is an important case to consider, and has the potential to confound several potential analyst tasks.

There are also a variety of visual design choices that can be made to enhance viewer understanding of the data and provide scaffolding for particular analysis tasks. These data attributes specify potential challenges in representing the data, which prompts particular sets of design decisions.

5 Design Decisions

To understand the breadth of the space of scatterplot designs, we collected designs and organized a taxonomy of design decisions for scatterplots (see Table 3). We posit that any scatterplot-like visualization will use some combination of these design variables in its construction. We identify these design variables through a separate literature survey (disconnected from §3). By enumerating these design decisions, we can use the previously-listed factors of analysis task and data characteristics to help determine the applicability of design decisions.

We collect design decisions from their use in visualization research papers. These decisions range in complexity from simple design decisions of the points (their color, size, and texture) to more advanced grouping techniques (convex hull shapes, KDE blending). By identifying these design decisions, we can start to identify strengths of different design strategies while also providing a framework in which to organize future techniques according to their task support. Combined with the list of scatterplot analysis tasks, the full framework (§3–5) can be used to link design variables with their task support, conditioned on the given characteristics of the data (see application in §6). We provide the full details about these sources, their design decisions, and which tasks and data characteristics are supported by each strategy in the supplementary material.

Relevant manuscripts were gathered through a keyword search methodology. We searched the titles and abstracts of articles published in the Information Visualization journal proceedings, EuroVis proceedings, Pacific Vis proceedings, and all VIS proceedings (SciVis/Vis, InfoVis, VAST) from 2009 to 2017 (3040 papers) for any instances of the string "scatter." Our query returned 117 results, of which 62 were relevant to scatterplots (a common matching element was "scattering," a component of rendering). We then perused these articles, pulling out information such as the anticipated support of scatterplot-specific tasks, the design strategy utilized, and the types of encodings evaluated or explicitly supported in the presented technique or experiment. This information is available within the supplementary material.

A benefit of building this space is that it articulates the range of scatterplots that different decisions make. This space thereby suggests potential programmer and designer abstractions that should support this range. To assist in realizing scaterplot designs in practice, we have developed a D3-based [7] library for scatterplot-like designs, called *d3-twodim* and available online at `https://uwgraphics.github.io/d3-twodim/`. The library allows programmers to experiment with designs utilizing both SVG and WebGL, adding automatic interaction support for linked components such as dropdown menus and tooltips.

5.1 Clustering of Design Choices

After collecting design choices (right-most column of Table 3), we group these choices together into clusters. Grouping these choices together helps to clarify the purpose and application for appropriate design. In the clustering, these decisions modify the design of the marks themselves (*point encoding*), group points by a visual technique (*point grouping*), modify the marks' position (*point position*), or add annotations, call-outs, or other amenities to the scatterplot (*graph amenities*). These clusters are discussed below, with some discussion of strategies that utilize these design decisions to support a particular analysis goal.

Point Encodings cover the design variables that can be applied to marks to represent objects in the graph, and can serve to differentiate encoded objects from one another. These types of common encodings can be considered as the decisions to be made on "marks", as many visualization grammars describe them (such as Wilkinson's graphics grammar [72]). Examples of these encodings are color, size, shape, and orientation. Careful use of these encodings can take advantage of pre-attentive processing of the human visual system [16, 71], directing the viewer's attention to particular subsets or patterns. Combinations of encodings can help viewers select subsets of points relevant to their exploration. Deliberate use of these encodings to group points together can be considered as an implicit grouping, which we discuss next.

Point Grouping decisions serve either to simplify the visual product by aggregating similar items together or to differentiate items from one another. Their role tends to further constrain and focus the overall message of the visualization. The term "grouping" is analogous to the usage of the term *abstraction* in visualization (see the use in Elmqvist and Fekete [26]). As we use it here, however, we consider grouping to be a superset of aggregation design decisions, and the choice of design strategy will emphasize a particular message. Design decisions under the point grouping designation drive task performance by narrowing the scope of potential insights—for example, collecting points into bins sacrifices the fidelity of item detail but exposes and highlights distributions of data.

Our framework organizes point grouping into *implicit* and *explicit* groupings. Implicit grouping uses *point encodings* to identify points as belonging to similar groups by categorization, distance in attribute value, or other similarity metric. Implicit strategies show data as points and rely on the viewer's perception to group points together, generally by way of gestalt grouping [71, 76]. In contrast, explicit grouping reduces object-specific fidelity to abstract marks and communicate aggregate, high-level judgments of the data. A canonical example in this space is binning [11], where group statistics of marks are collected for small, regular spatial regions—while this trades off the fidelity of individual marks for more aggregate judgments, it may be better able to communicate the numerosity differences in different regions in the plot.

Polygon enclosure, by continuous, majority, or full convex hull means, can simplify areas of high visual noise to indicate highly dense regions. For example, VisIRR [15] uses a simplified ellipse to collect groups of points together while also redundantly encoding category with color. In particular, these strategies can be composited with point position strategies to explicitly support a small set of analysis tasks, similar to the strategy of Splatterplots [49]—group marks together, then explicitly restore and style individual marks that fall outside the grouped region. These enclosures need not be enclosed shapes; Cleveland and McGill [18] use smoothings (upper/lower residuals) to emphasize the correlation of the two axes. Shape abstraction can also be used to emphasize the trend of a distribution, such as used by Yates *et al.* [74] to emphasize different logical implications between the two axes of a scatterplot.

Point Position — Scatterplots tend to display data items by creating a spatialization by two continuous attributes. However, some designs modify point position to pack more information into the visualization (e.g. reducing dimensionality) or emphasize particular areas of the graph (zooming and displacement) These decisions are made to emphasize support for particular tasks over absolute accuracy and faith to the original data. By modifying point position, these strategies combat issues of overdraw stemming from either a poor distribution of data or simply coping with the inevitable overdraw with too many objects for the screen-space. Utilizing these strategies can help to emphasize distributional judgments, assist in identifying and tracking objects of interest, reduce more than two dimensions to a familiar scatterplot design, and visually organize data for subsequent decisions (both point encoding and grouping strategies). As an example, Chen *et al.* [14] use "smart" subsampling to convey distributions of multiple series while minimizing overdraw of individual marks. Keim *et al.* [37] use a subspace warp to effectively use unneeded screen-space to emphasize

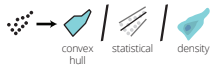
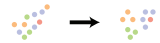
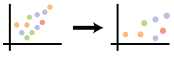
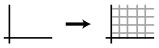
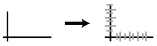
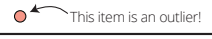
Cluster	Design Choice	Example
Point Encoding	Color	
	Size	
	Symbols	
	Outline	
	Opacity	
	Texture	
	Depth of Field	
	Blurriness	
Point Grouping	Representation Type	
	Positional Binning	
	Polygon Enclosure	
	Shape Abstraction	
Point Position	Subsampling	
	Displacement	
	Animation	
	Projection	
	Zooming	
Graph Amenities	Grid Lines	
	Axis Ticks	
	Legend	
	Trend Lines	
	Annotations	

Table 3. A categorization of design decisions available to the scatterplot designer, which are clustered into four categories. Each of these categories can be used to gauge appropriate design strategies.

distribution judgments.

Graph Amenities — Annotations and other scaffolding can help the viewer navigate a scatterplot. Examples of these strategies include grid lines, axis ticks, object labeling, encoding legends, and trendlines. These amenities help by orienting the viewer (e.g., axis ticks) and providing additional information (e.g., legends, sensitivity lines) relevant to analysis. Much like point position strategies, scaffolds of this type can serve to emphasize the particular message of the visualization, specifically helping viewers to complete object-centric tasks. Most critically, these amenities can help a viewer navigate the visualization by highlighting relevant items through annotation, provide distributional context with tick lines, and highlight potential correlations with trend lines.

5.2 Interaction Intents

Interaction commonly accompanies scatterplots to support the intentions of viewers. While interactions are not necessarily *visual* design decisions, they are commonly used in conjunction with visual strategies to support the tasks of the viewer. Brehmer and Munzner [8] also motivate the inclusion of interaction intents as the "how" in their task typology—a critical component to support changing the visual strategy to support viewers in their analysis. In the same vein as Amar *et al.* [2], these intents signify the desire of the viewer to change the granularity of the visualization or change the reference frame. These intents indicate a desire to directly contrast or evolve the current set of design decisions with a new set, incorporating the strategies that will make the appropriate design variable changes to support the given intent. With a change in the design, the spectrum of task support changes—potentially in a deliberate way.

Interaction can signal that the viewer wants to change to a view that is more appropriate for their desired analysis task. For example, a viewer may want to *focus analysis on relevant items*, in which case they may be able to interact with items in the visualization (direct interaction), brushing (selection), or by interacting with a linked component (e.g., text-box, table of attributes). To emphasize the relevant objects or groups, a *point encoding* could be assigned to highlight the relevant marks. Two common pivots deal with changing the level of granularity—seeing more detail or less detail; "elaborating" and "summarizing" by the taxonomy of Schulz *et al.* [57]. *Seeing more detail* could involve actions such as zooming or jittering, both examples of *point position* design strategies. *Seeing less detail* could involve abstraction through subsampling or aggregation, examples of *point grouping* strategies.

Thinking about interaction as an intent to change the visual design to support a competing task can help rationalize the controls that a viewer has. For example, InterAxis [38] allows viewers to use exemplar objects to dynamically weight and re-project the dataset to identify related objects, allowing viewers to change their frame of reference. Many lensing techniques, such as MoleView by Hurter *et al.* [35], use the lens to select relevant types of items as a way of reducing distraction from other overlapping elements. MoleView also supports aggregation behavior (such as edge bundling) within the lens to further reduce element complexity. This highlights an intent from the viewer to switch from a high-level overview of the data toward a more localized, detailed neighborhood exploration setting. These intents provide another layer of abstraction to group design decisions for supporting analysis tasks.

6 Linking Areas of the Space

Our framework suggests that scatterplot designs should be matched to the tasks and data characteristics that they are designed to support. The tasks and data characteristics form a high-dimensional space—any scenario is a point in this space. For any one point in the space, we can determine which design decisions are appropriate. Creating a map of this entire space is challenging because it is large. Even if we divide the axes into discrete buckets (such as §4), we are left with 12 (tasks) $\times$ 4 (points) $\times$ 3 (dims) $\times$ 2 (spatial) $\times$ 5 (distribution), yielding a grid of over 4300 discrete scatterplot scenarios.

For each scenario, we seek to determine which of the five design cluster strategies are appropriate. In some cases, this will be easy to determine. There may be examples that prove the effectiveness of a design strategy for a scenario, or reasoning about factors can determine inappropriateness (e.g., point encodings are inappropriate for identifying an object among millions of points). In other cases, however, the decision may not be so clear: it may require an empirical study to determine if a design is effective for a scenario; there is the potential for a specific novel design that effectively employs the strategy for a scenario; or the strategy is only effective under certain circumstances.

The massive grid of effectiveness decisions would be attractive, but also infeasible to fully realize because of its size. Additionally, many entries would only be our current subjective assessment subject to change based on newly discovered designs or empirical evidence. Furthermore, presenting this high-dimensional grid as figures in this paper would be challenging. For these reasons, we have not attempted to provide the full table. Instead, we have given our (current, subjective) assessments for a large portion of the grid as supplemental data and also provide a web-based tool for exploring a different slice of this

high-dimensional table[1]. We show a representative "slice" of the table below, showing how our framework can be used to match scatterplot designs to analysis scenarios.

While a pre-determined table of appropriateness would be convenient, our framework can be applied without it. The important part of the framework is that it enumerates the factors to consider and the design choices—informing the structure of the grid. Specific assessments of appropriateness should be the subjective opinion of the designer based on the concerns detailed in Sections 3–5. Examples of applying this type of analysis for a range of scenarios is provided in the next section.

6.1 A Slice of the Space: Tasks and Design Strategies

We illustrate our framework with a small slice of the entire grid: a specific set of data characteristics, the entire range of tasks, and the entire set of design strategies. For the sake of demonstration of the framework and to support discussion of the current high-level trends and strategies in scatterplot design, we are providing 60 out of the 4300 cells of the overall table. To demonstrate an interesting reference point where the design of a faceless scatterplot becomes intractable for many tasks, we choose a particular set of data characteristics. This slice *fixes the set of data characteristics* to a moderate number of objects and number of classes, in an *unstructured* distribution of scattered data. We note that we could also take an alternative slice of the map, with 10 points, no class label, in a random distribution, and the map would provide a wildly different set of appropriateness measures.

With this map and aforementioned slice in particular, we examine how and why certain encoding decisions can or cannot support particular analysis tasks. As an example, identifying and comparing numerosity in a faceless scatterplot can start to become challenging when many points are overlapping, masking the viewer's determination of density (and thereby suggesting a design change). In Table 4 above, we denote appropriate design decisions with a ✔, potential design support with ✔*, support possible with an accompanying design decision with ✧, and inappropriate support with ✘. These determinations are made and motivated by our assessments of the state-of-the-art, existence proofs of design and interaction techniques in the research literature (informed by our survey detailed in §5), and empirical experimentation of encoding decisions for specific viewer tasks. In the prose below, we describe specific decisions in the slice and describe their extrapolation to the broader table. We also contrast suggested designs with designs that may work better in other scenarios with different data characteristics.

At a high-level, appropriateness for design decisions for various tasks begins to expose clusters of similarly-supported tasks. The cells in the slice are referenced by the task (a number) and the encoding type (a representing letter). We discuss some of the short-comings of typical strategies for scatterplot design, and provide pointers to exemplar systems that can scaffold the desired analysis tasks.

- *Difficult to support aggregate-level tasks with point encodings* (**9A–11B**) — Tasks 9, 10, and 11 deal with aggregate-level tasks that seek to uncover characteristics about the data on a global scale, either by identifying those marks that are outliers or anomalies, gauging correlation across the dataset, or understanding object density across the graph area. Due to the aggregate nature of these tasks, utilizing the strategy of how marks are encoded (**A**) or moving point positions (**B**) will not help. The similarity of encoding strategy effectiveness among these three tasks suggest that it may be fruitful consider these three tasks under an "aggregate-level" umbrella, where encoding decisions made to support these tasks stand in opposition to "object-level" tasks. In scenarios with fewer points, it may be possible to support these tasks with implicit grouping. However, such approaches would not apply in situations with significant overdraw.

- *Unclear how to design interaction and amenities for aggregate-level tasks* (**10D, 11D–E**) — There is a clear gap in designing interactions (**D**) for aggregate-level tasks such as identifying correlation (#10) or performing comparisons in object numerosity (#11). Direct manipulation approaches have been proposed [56], though exactly how to prompt viewers to interact with the visualization to promote correlation or numerical understanding is unclear. While graph amenities (**E**) can help to see correlation (such as overlaying a trend line over the data), identifying and comparing numerosity in multiple areas on the plot becomes difficult with many annotations and call-outs. To potentially address these issues, landscape views [68] use point grouping strategies to emphasize numerosity judgments.

[1] http://graphics.cs.wisc.edu/Vis/scattertasks

Task	A Point encoding	B Point position	C Point grouping	D Interaction intent	E Graph amenities
1 Identify object	✔	✔	✧	✔	✔*
2 Locate object	✔	✧	✧	✔	✔
3 Verify object	✔	✔*	✧	✔	✔
4 Compare objects	✔	✔	✧	✔	✔
5 Explore neighborhood	✔	✔	✔	✔	✔
6 Search for motif	✔	✔	✔	✔	✔*
7 Explore data	✔	✔	✔	✔	✔
8 Charact. distribution	✔	✔	✔	✧	✔
9 Find anomalies	✧	✔*	✧	✔*	✔
10 Identify correlation	✘	✘	✔	✘	✔
11 Charact. numerosity	✘	✘	✔	✘	✘
12 Charact. distances	✔*	✔	✔*	✔*	✔

Table 4. A 2D slice of the task support map by clusterings of visual encodings, with data characteristics set to a "large" number of points with a few number of classes in a non-clustered position (so the possibility of overdraw exists). ✔ denotes general support, ✔* denotes support in particular situations (discussed in prose), ✧ requires concurrent support from other encodings, while ✘ identifies no improvement to task support.

- *Losing mark fidelity with* ***point grouping*** (**1C–4C, 9C, 12C**) — Point grouping (**C**) provides a way to abstract and convey a particular narrative about the data. By aggregating marks into large visual shapes, designs using point grouping strategies lose the support of object-centric tasks such as finding outliers and comparing objects. As an example, performing continuous aggregation via KDE [58] would support judgments of comparing numerosity across the plot (**C11**), but would not support object-centric tasks such as *locate object* (**C2**).

  However, by compositing aggregation operations with point encodings, point positions, and interaction intents, object-centric tasks can be supported. As an example, an interaction where a viewer hovers over a filled-in region could subsequently highlight exemplar points, which could then be explicitly selected for object comparison (#4). Many scatterplot-like techniques use a composition to restore support for object-centric tasks, such as Splatterplots [49] and Chen *et al.*'s sampling strategy [14]. Exactly what design patterns that may prompt a viewer or an analyst to engage with an aggregated display to perform an object-centric task remains an open question, though many systems use interactions such as brushing to populate an external component, such as a "selected" list.

For specific concerns in Table 4, there exist several classes of design strategies that can help bolster the efficacy of analytical tasks.

- *Supporting distance judgments* (**12A–D**) — The distance between marks (task #12) takes on a different meaning based on the dimensionality being visualized. The marks may be placed based on two continuous attributes of the objects, where the distance between marks

communicates the distance in attribute space, or the marks could be placed based on two dimensionally-reduced, derived dimensions, where data is placed based on the total similarity of its continuous attributes. To support the analysis of dimensionally-reduced data in a scatterplot, many visual analytics systems provide scaffolding by amenities or external linked components. Dis-Function [10], for example, supports direct interaction of individual marks to update the similarity projection of the entire high-dimensional dataset.

- *Dealing with overdraw* (**1E, 6E, 9D**) — With significant data, the possibility of overdraw or masking of object-representing marks exists. This hurts detection of individual points, and designs have been constructed to preserve judgments of numerosity [21, 37, 49, 68] or use alternative methods such as visual aggregation [30] to preserve statistical judgments. Many of these designs do not use graph amenities (**1E**). However, paired with a lensing technique (see generally Tominski *et al.* [67]), this analysis scenario could be supported. Similarly, exposing a given distributional motif (**6E**) is difficult given that this motif may not be known to the visualization designer *a priori*—but specialized amenity techniques such as drawing moment lines [13] can convey an aggregate sense of a motif. With increased numbers of points, however, these amenities can themselves exacerbate the problem of overdraw.

  Again in an overdraw scenario, it may be difficult to distinguish outliers or anomalies with an interaction intent (**9D**)—how might an analyst specify "show me the outliers" directly within the plot? One strategy is to compose strategies with other operations: Splatterplots [49] explicitly selects those marks that fall outside thresholded density regions, and ensures those marks are visible while zooming the plot.

- *Consciously supporting object-centric tasks* (**1C–4C, 9C, 2B**) — Marks that represent objects are needed to obtain information about individual objects. Object-specific tasks (#1–3) and object-centric tasks (#4, 9), such as compare objects, depend on the specific marks for a viewer to perform their desired analysis, but many point grouping techniques (**C**) aggregate marks together. To be able to support these tasks, several different types of strategies have been developed; a common strategy to support these object-centric tasks is to provide a external filtering component that selects objects based on semantic content or viewer-defined thresholds, then highlights the selected objects as marks overlaying the aggregate encodings. This strategy can also help finding the positions of marks if the points are moved (**2B**). Many interactive lensing techniques have also been developed, where a viewer can mouse-over to see more detail of the objects contained within the lens scope [34, 67].

The supplemental material provides a listing of 62 strategies that handle the analysis scenarios raised within this linking table, organized by the characteristics of data supported, the analysis tasks supported, and the types of design decisions used. We illustrate common themes in scatterplot design in the discussion section.

7 Discussion

Throughout the paper, we have developed a framework to discuss the design of scatterplots. Using the task list (§3), we are able to focus our attention on how those tasks are supported by scatterplot designs and affected by characteristics of the data. Trends of task support by data characteristics for traditional scatterplots have been identified, and lead to suggestions of design strategies to support the desired tasks. These suggestions lead to trade-offs in the design of scatterplots. There are instances in scatterplot design where the circumstances of the data prevent a single design strategy from supporting all tasks. For example, a density-based encoding with thousands of points can support the task of numerosity comparison easily, but needs conscious design support for identifying outliers.

The following themes highlight potential challenges in designing effective scatterplots, and suggest strategies for supporting common analysis scenarios.

Visual Complexity / Too Many Points — Dealing with visual clutter has been the focus of many visualization techniques and taxonomies. In particular, Ellis and Dix [25] explore a wide range of strategies and the trade-offs between them. Many of the techniques that we found through our literature search employed some method of visual simplification, explicitly supporting some analysis tasks while weakening support for others. These strategies generally fall under the categories of point grouping and point position strategies. Point grouping strategies generally abstract groups of points into fewer distinct visual structures, emphasizing numerosity and distributional judgments at the expense of tasks dealing with individual objects. Through the point grouping process, however, the ability to identify both outliers and anomalies usually becomes hindered (aggregate-level tasks).

On the other hand, point position strategies such as projection and animation can pack additional structural information into a scatterplot without sacrificing the viewer's ability to execute element-specific tasks. While these methods necessarily modify the "true state" of each mark's spatialization, these methods can emphasize hidden or overlapping structure based on the characteristics of the data. As an example, generalized scatter plots [37] warp the subspace of the plot area to maximize the use of space (point position) and utilize a KDE-like point grouping strategy to emphasize the numerosity of points.

A common problem in scatterplots is the problem of overdraw when there are simply too many marks for the available chart area. Similar to the visual complexity problem, both grouping and position strategies can help alleviate the issues of incomprehensibility at scale. A generalized set of point grouping strategies provide different levels of support for analysis tasks. In principle, the plan of what features of the data to communicate determines the scope of design strategies that emphasize those characteristics.

Demonstrating *distributions* is well-supported by density-driven encodings, such as shape binning [11] or continuous density estimation [58]. By abstracting away individual point marks and using visual weight to communicate relative numerosity, we can support the aggregate-level tasks such as characterize distribution or identify correlation at the expense of object-centric tasks such as object comparison or verify object. While examples of these density-driven encodings are numerous, there are particular design details within these strategies that have trade-offs of support between the scatterplot analysis tasks.

Effectively communicating *numerosity* can often be concurrently supported by strategies that emphasize distribution, though caveats exist. For strategies that support kernel density estimation [37, 49], a thresholded region may communicate the range of a high-number of points, but without a complex contour map [19], it can be difficult to compare approximate number of points. Aggregation commonly has computational complexity on the order of the number of points, though some (such as Splatterplots) may use the GPU to compute repetitive density estimation. Computationally simpler strategies can utilize blur [63] or alpha encodings [21] to communicate relative numerosity of marks, given an appropriate normalization dependent on the current view [48].

Figure 3 shows a side-by-side comparison of three scatterplot designs, all displaying the same dataset with a "medium" number of points—individual points can be discerned, and class distribution is still apparent in a faceless scatterplot. However, not all tasks are equally supported by each design—the faceless scatterplot supports object-centric tasks (#1–3) with some overdraw, while colored contour maps [19] (center) eschew object-centric tasks to focus attention on distributions and densities. Comparatively, the Splatterplot [49] (right) shows outlier points, but aggregates points together using a thresholded KDE, providing a sense of locality of dense regions between the classes. While both the contour map and Splatterplot use point grouping strategies, the contour map provides more information about density information than the Splatterplot—which could sway a designer's choice of design strategy depending on the analysis goals of the viewer.

Differentiating Groups of Marks / Too Many Classes — Many strategies have been proposed to differentiate groups of marks. Much early work has concentrated on the perceptual grouping of points, with Cle-

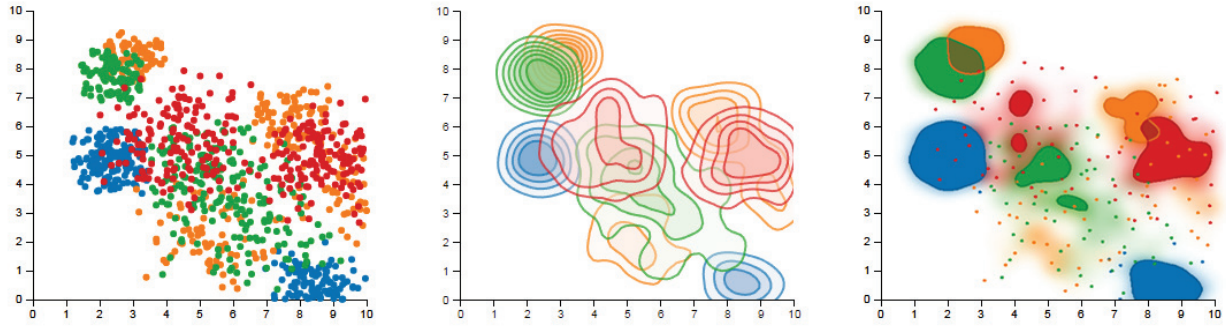


Fig. 3. Three different designs (left-to-right: traditional scatterplot, contour map [19], and Splatterplot [49]) display different information about the same 100 item, four class (mapped to color) dataset. While the traditional scatterplot exhibits some overdraw, the two alternative approaches use point grouping techniques to emphasize numerosity and distribution comparison tasks. The contour map conveys density gradients, while the Splatterplot uses thresholded regions to convey dense areas.

veland [16] mentioning ways of emphasizing groups of points by using distinct encodings. Mackinlay [47] provides a perceptual ordering of encoding decisions, Ware [71] describes the perceptual basis behind the ordering of the visual variables, and Li *et al.* explores perceptual sensitivity to these factors in scatterplot applications [43–45]. Using point encodings to separate marks into groups is a very common trait, usually to split data into separate series or categories. While supporting object-centric tasks such as *locate object* and *identify anomalies*, these type of solutions also promote the exploration of data by creating interesting structures in the data to peruse.

An open problem in scatterplot design is how to communicate large numbers of series or categorization for marks. In many analysis scenarios, the number of classes to consider may number from the tens to hundreds of classifications, where comparison in numerosity or distribution between any number of series may be important to the analysis. A core limiting factor is the number of encodings to use to distinguish marks from each other: color has a fidelity of around 12 distinct hues [71], which rapidly declines with smaller visual area [64]. Different shapes can also provide additional separation, but again suffer at small sizes. Some strategies allow the viewer to focus on a small subset of series and place all other data into a "background" group [39, 63], or take advantage of hierarchy within the data to group similar objects together [26]. The literature lacks techniques for handling large numbers of classes, even though the problem is common, often appearing in humanities analysis contexts [1, 34].

Communicating High-Level Statistics — In many scenarios, it may be advantageous to communicate the distribution of the data or highlight potential correlation. Studies such as those by Gleicher *et al.* [30] have shown how encoding decisions can affect viewer judgments of group statistics without explicit representation by graph amenities or point grouping (such as the smoothings as presented by Cleveland and McGill [18]). While it may be important to explicitly support statistics of the data through graph amenities (e.g. annotations or showing a confidence interval), supporting statistical judgments implicitly can help in analyses where the specific statistics important for analyses are not known *a priori*. Some designs use shape aggregation to emphasize distributions, such as pictograms by Lehmann *et al.* [41] or glyph SPLOMs by Yates *et al.* [74], sacrificing object-level judgments for rapid distribution judgments.

Too Many Dimensions — Pragmatically, the number of dimensions should not affect the appearance of a scatterplot, as only two dimensions are shown. However, tasks performed with dimensionally-reduced or projected data tend to differ from the tasks done on two-dimensional data. To this end, many dimensionally-reduced scenarios contain extra detail about objects and can permit direct manipulation to feed back into the dimension-reduction algorithm. Strategies such as DisFunction [10] or InterAxis [38] use direct viewer interaction to drive the semantic clustering of similar objects together. To support visualizing multiple dimensions without precomputation, multi-axis embeddings such as star coordinates [36] or their orthographic variant [42] can expose clusters in a two-dimensional embedding. Many of these scenarios concentrate on object-centric and distributional scenarios that highlight the semantic similarity between objects.

8 Conclusion

Scatterplots are a visualization design widely applicable to a large range of analysis scenarios. With the many different design strategies available to select from, understanding the trade-offs between the many design choices is challenging. In this work, we have introduced a framework to help determine the design appropriateness for task support, and show how this framework can help gauge task performance dependent on characteristics of the data. With the characterization of this design space, we have described the challenges, existing solutions for these challenges, and potential areas for innovation in scatterplot design.

Acknowledgments

The authors would like to thank the anonymous reviewers for their actionable feedback, Eric Alexander, Danielle Szafir, and Michael Correll for fruitful discussion, and Robin Valenza for copy-editing. This work was supported by NSF award IIS-1162037.

References

[1] E. Alexander and M. Gleicher. Task-driven comparison of topic models. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):320–329, 2016. doi: 10.1109/TVCG.2015.2467618

[2] R. Amar, J. Eagan, and J. Stasko. Low-level components of analytic activity in information visualization. In *IEEE Symposium on Information Visualization*, pp. 111–117. IEEE, 2005. doi: 10.1109/INFVIS.2005.1532136

[3] S. Bachthaler and D. Weiskopf. Continuous scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1428–1435, 2008. doi: 10.1109/TVCG.2008.119

[4] E. Bertini and G. Santucci. Give chance a chance: Modeling density to enhance scatter plot quality through random data sampling. *Information Visualization*, 5(2):95–110, 2006. doi: 10.1057/palgrave.ivs.9500122

[5] E. Bertini, A. Tatu, and D. Keim. Quality metrics in high-dimensional data visualization: An overview and systematization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2203–2212, 2011. doi: 10.1109/TVCG.2011.229

[6] L. Best, A. Hunter, and B. Stewart. Perceiving relationships: A physiological examination of the perception of scatterplots. *Diagrams*, pp. 244–257, 2006. doi: 10.1007/11783183_33

[7] M. Bostock, V. Ogievetsky, and J. Heer. D^3: Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011. doi: 10.1109/TVCG.2011.185

[8] M. Brehmer and T. Munzner. A multi-level typology of abstract visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2376–85, 2013. doi: 10.1109/TVCG.2013.124

[9] M. Brehmer, M. Sedlmair, S. Ingram, and T. Munzner. Visualizing dimensionally-reduced data: Interviews with Analysts and a Characterization of Task Sequences. In *Proc. Beyond Time and Errors Novel Evaluation Methods for Visualization (BELIV '14)*, pp. 1–8. ACM Press, New York, New York, USA, 2014. doi: 10.1145/2669557.2669559

[10] E. T. Brown, J. Liu, C. E. Brodley, and R. Chang. Dis-Function: Learning distance functions interactively. In *IEEE Conference on Visual Analytics Science and Technology*, pp. 83–92. IEEE, 2012. doi: 10.1109/VAST.2012.6400486

[11] D. B. Carr, R. J. Littlefield, W. L. Nicholson, and J. S. Littlefield. Scatterplot matrix techniques for large N. *Journal of the American Statistical Association*, 82(398):424, 1987. doi: 10.2307/2289444

[12] S. M. Casner. Task-analytic approach to the automated design of graphic presentations. *ACM Transactions on Graphics*, 10(2):111–151, 1991. doi: 10.1145/108360.108361

[13] Y.-H. Chan, C. D. Correa, and K.-L. Ma. Flow-based scatterplots for sensitivity analysis. In *IEEE Symposium on Visual Analytics Science and Technology*, pp. 43–50. IEEE, 2010. doi: 10.1109/VAST.2010.5652460

[14] H. Chen, W. Chen, H. Mei, Z. Liu, K. Zhou, W. Gu, and K.-L. Ma. Visual abstraction and exploration of multi-class scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1683–1692, 2014. doi: 10.1109/TVCG.2014.2346594

[15] J. Choo, C. Lee, H. Kim, H. Lee, Z. Liu, R. Kannan, C. D. Stolper, J. Stasko, B. L. Drake, and H. Park. VisIRR: Visual analytics for information retrieval and recommendation with large-scale document data. *IEEE Conference on Visual Analytics Science and Technology*, 1(C):243–244, 2015. doi: 10.1109/VAST.2014.7042511

[16] W. S. Cleveland. *The Elements of Graphing Data*. Wadsworth Advanced Books and Software, Monterey, CA, USA, 1985.

[17] W. S. Cleveland, M. E. McGill, and R. McGill. The shape parameter of a two-variable graph. *Journal of the American Statistical Association*, 83(402):289, 1988. doi: 10.2307/2288843

[18] W. S. Cleveland and R. McGill. The many faces of a scatterplot. *Journal of the American Statistical Association*, 79(388):807–822, 1984. doi: 10.2307/2288711

[19] C. Collins, G. Penn, and S. Carpendale. Bubble Sets: Revealing set relations with isocontours over existing visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1009–1016, 2009. doi: 10.1109/TVCG.2009.122

[20] M. Correll and M. Gleicher. What shakespeare taught us about text visualization. In *The 2nd Workshop on Interactive Visual Text Analytics*, 2012.

[21] J. Cottam, A. Lumsdaine, and P. Wang. Overplotting: Unified solutions under abstract rendering. In *IEEE International Conference on Big Data*, pp. 9–16. IEEE, 2013. doi: 10.1109/BigData.2013.6691712

[22] Q. Cui et al. Measuring data abstraction quality in multiresolution visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):709–716, 2006. doi: 10.1109/TVCG.2006.161

[23] T. N. Dang and L. Wilkinson. ScagExplorer: Exploring scatterplots by their scagnostics. In *IEEE Pacific Visualization Symposium*, pp. 73–80. IEEE, 2014. doi: 10.1109/PacificVis.2014.42

[24] M. Elliott and R. Rensink. Interference in the Perception of Two-Population Scatterplots. *Journal of Vision*, 15(12):893, 2015. doi: 10.1167/15.12.893

[25] G. Ellis and A. Dix. A Taxonomy of Clutter Reduction for Information Visualisation. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1216–1223, 2007. doi: 10.1109/TVCG.2007.70535

[26] N. Elmqvist and J. D. Fekete. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE Transactions on Visualization and Computer Graphics*, 16(3):439–454, 2010. doi: 10.1109/TVCG.2009.84

[27] J.-D. Fekete and C. Plaisant. Interactive information visualization of a million items. In *IEEE Symposium on Information Visualization*, pp. 117–124, 2002. doi: 10.1109/INFVIS.2002.1173156

[28] M. Fink, J. H. Haunert, J. Spoerhase, and A. Wolff. Selecting the aspect ratio of a scatter plot based on its delaunay triangulation. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2326–2335, 2013. doi: 10.1109/TVCG.2013.187

[29] 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. doi: 10.1002/jhbs.20078

[30] M. Gleicher, M. Correll, C. Nothelfer, and S. Franconeri. Perception of average value in multiclass scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2316–2325, 2013. doi: 10.1109/TVCG.2013.183

[31] C. C. Gramazio, K. B. Schloss, and D. H. Laidlaw. The relation between visualization size, grouping, and user performance. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1953–1962, dec 2014. doi: 10.1109/TVCG.2014.2346983

[32] C. G. Healey, K. S. Booth, and J. T. Enns. High-speed visual estimation using preattentive processing. *ACM Transactions on Computer-Human Interaction*, 3(2):107–135, 1996. doi: 10.1145/230562.230563

[33] J. Heer and M. Agrawala. Multi-scale banking to 45. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):701–708, 2006. doi: 10.1109/TVCG.2006.163

[34] F. Heimerl, M. John, Q. Han, S. Koch, and T. Ertl. DocuCompass: Effective exploration of document landscapes. In *IEEE Conference on Visual Analytics Science and Technology*, pp. 11–20, 2016. doi: 10.1109/VAST.2016.7883507

[35] C. Hurter, O. Ersoy, and A. Telea. MoleView: An attribute and structure-based semantic lens for large element-based plots. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2600–2609, 2011. doi: 10.1109/TVCG.2011.223

[36] E. Kandogan. Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In *ACM International Conference on Knowledge Discovery and Data Mining*, pp. 107–116. ACM Press, New York, New York, USA, 2001. doi: 10.1145/502512.502530

[37] D. A. Keim, M. C. Hao, U. Dayal, H. Janetzko, and P. Bak. Generalized scatter plots. *Information Visualization*, 9(4):301–311, 2010. doi: 10.1057/ivs.2009.34

[38] H. Kim, J. Choo, H. Park, and A. Endert. InterAxis: Steering scatterplot axes via observation-level interaction. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):131–140, 2016. doi: 10.1109/TVCG.2015.2467615

[39] R. Kincaid and K. Dejgaard. MassVis: Visual analysis of protein complexes using mass spectrometry. In *IEEE Symposium on Visual Analytics Science and Technology*, pp. 163–170, 2009. doi: 10.1109/VAST.2009.5333895

[40] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry. Task taxonomy for graph visualization. In *Proc. 2006 AVI BELIV*, pp. 1–5. ACM Press, New York, New York, USA, 2006. doi: 10.1145/1168149.1168168

[41] D. J. Lehmann, F. Kemmler, T. Zhyhalava, M. Kirschke, and H. Theisel. Visualnostics: Visual guidance pictograms for analyzing projections of high-dimensional data. *Computer Graphics Forum*, 34(3):291–300, 2015. doi: 10.1111/cgf.12641

[42] D. J. Lehmann and H. Theisel. Orthographic star coordinates. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2615–2624, 2013. doi: 10.1109/TVCG.2013.182

[43] J. Li, J.-B. Martens, and J. J. van Wijk. Judging correlation from scatterplots and parallel coordinate plots. *Information Visualization*, 9(1):13–30, 2008. doi: 10.1057/palgrave.ivs.9500179

[44] J. Li, J.-B. Martens, and J. J. van Wijk. A model of symbol size discrimination in scatterplots. In *Proc. Conference on Human Factors in Computing Systems*, p. 2553. ACM Press, New York, New York, USA, 2010. doi: 10.1145/1753326.1753714

[45] J. Li, J. J. van Wijk, and J.-B. Martens. A model of symbol lightness discrimination in sparse scatterplots. In *IEEE Pacific Visualization Symposium*, pp. 105–112, 2010. doi: 10.1109/PACIFICVIS.2010.5429604

[46] A. M. MacEachren. *How Maps Work: Representation, Visualization, and Design*. The Guilford Press, New York, New York, USA, 1995.

[47] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, 5(2):110–141, 1986. doi: 10.1145/22949.22950

[48] J. Matejka, F. Anderson, and G. Fitzmaurice. Dynamic opacity optimization for scatter plots. In *Proc. Conference on Human Factors in Computing Systems*, pp. 2707–2710. ACM Press, New York, New York, USA, 2015. doi: 10.1145/2702123.2702585

[49] A. Mayorga and M. Gleicher. Splatterplots: Overcoming overdraw in scatter plots. *IEEE Transactions on Visualization and Computer Graphics*, 19(9):1526–1538, 2013. doi: 10.1109/TVCG.2013.65

[50] D. R. Montello, S. I. Fabrikant, M. Ruocco, and R. S. Middleton. Testing the first law of cognitive geography on point-display spatializations. *Cosit*, pp. 316–331, 2003. doi: 10.1007/978-3-540-39923-0_21

[51] T. Munzner. *Visualization Analysis & Design*. CRC Press: Taylor & Francis Group, Boca Raton, FL, 2014.

[52] R. Rensink and G. Baldridge. The perception of correlation in scatterplots.

Computer Graphics Forum, 29(3):1203–1210, 2010. doi: 10.1111/j.1467-8659.2009.01694.x

[53] R. A. Rensink. The nature of correlation perception in scatterplots. *Psychonomic Bulletin & Review*, 24(3):776–797, 2017. doi: 10.3758/s13423-016-1174-7

[54] A. Rind, W. Aigner, M. Wagner, S. Miksch, and T. Lammarsch. Task Cube: A three-dimensional conceptual space of user tasks in visualization design and evaluation. *Information Visualization*, 15(4):288–300, 2016. doi: 10.1177/1473871615621602

[55] R. E. Roth. An empirically-derived taxonomy of interaction primitives for interactive cartography and geovisualization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2356–2365, 2013. doi: 10.1109/TVCG.2013.130

[56] B. Saket, H. Kim, E. T. Brown, and A. Endert. Visualization by demonstration: An interaction paradigm for visual data exploration. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):331–340, 2017. doi: 10.1109/TVCG.2016.2598839

[57] H. J. Schulz, T. Nocke, M. Heitzler, and H. Schumann. A design space of visualization tasks. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2366–2375, 2013. doi: 10.1109/TVCG.2013.120

[58] D. W. Scott. Kernel density estimators. In *Multivariate Density Estimation*, pp. 125–193. John Wiley & Sons, Inc., 2008. doi: 10.1002/9780470316849.ch6

[59] M. Sedlmair, T. Munzner, and M. Tory. Empirical guidance on scatterplot and dimension reduction technique choices. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2634–2643, 2013. doi: 10.1109/TVCG.2013.153

[60] M. Sedlmair, A. Tatu, T. Munzner, and M. Tory. A Taxonomy of Visual Cluster Separation Factors. *Computer Graphics Forum*, 31(3pt4):1335–1344, 2012. doi: 10.1111/j.1467-8659.2012.03125.x

[61] M. Sips, B. Neubert, J. P. Lewis, and P. Hanrahan. Selecting good views of high-dimensional data using class consistency. *Computer Graphics Forum*, 28(3):831–838, 2009. doi: 10.1111/j.1467-8659.2009.01467.x

[62] D. Spencer. *Card Sorting: Designing Usable Categories*. Rosenfield Media, Brooklyn, New York, 2009.

[63] J. Staib, S. Grottel, and S. Gumhold. Enhancing Scatterplots with Multi-Dimensional Focal Blur. *Computer Graphics Forum*, 35(3):11–20, 2016. doi: 10.1111/cgf.12877

[64] M. Stone, D. A. Szafir, and V. Setlur. An engineering model for color difference as a function of size. In *IS&T 22nd Color Imaging Conference*, pp. 253–258, 2014.

[65] J. Talbot, J. Gerth, and P. Hanrahan. Arc length-based aspect ratio selection. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2276–2282, dec 2011. doi: 10.1109/TVCG.2011.167

[66] A. Tatu, P. Bak, E. Bertini, D. Keim, and J. Schneidewind. Visual quality metrics and human perception. In *Proceedings of the International Conference on Advanced Visual Interfaces*, pp. 49–56. ACM Press, New York, New York, USA, 2010. doi: 10.1145/1842993.1843002

[67] C. Tominski, S. Gladisch, U. Kister, R. Dachselt, and H. Schumann. Interactive lenses for visualization: An extended survey. *Computer Graphics Forum (STAR)*, (00):1–28, 2016. doi: 10.1111/cgf.12871

[68] 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. doi: 10.1109/TVCG.2007.70596

[69] Tuan Nhon Dang, L. Wilkinson, and A. Anand. Stacking graphic elements to avoid over-plotting. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1044–1052, 2010. doi: 10.1109/TVCG.2010.197

[70] D. K. Urribarri and S. M. Castro. Prediction of data visibility in two-dimensional scatterplots. *Information Visualization*, 16(2):113–125, 2017. doi: 10.1177/1473871616638892

[71] C. Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann, Burlington, MA, 3rd ed., 2012.

[72] L. Wilkinson. *The Grammar of Graphics (Statistics and Computing)*. Spinger Science+Business Media, Inc., New York, New York, USA, 2nd ed., 2005.

[73] L. Wilkinson, A. Anand, and R. Grossman. Graph-theoretic scagnostics. In *IEEE Symposium on Information Visualization*, pp. 157–164. IEEE, 2005. doi: 10.1109/INFVIS.2005.1532142

[74] A. Yates, A. Webb, M. Sharpnack, H. Chamberlin, K. Huang, and R. Machiraju. Visualizing multidimensional data with glyph SPLOMs. *Computer Graphics Forum*, 33(3):301–310, 2014. doi: 10.1111/cgf.12386

[75] X. Yuan, D. Ren, Z. Wang, and C. Guo. Dimension Projection Matrix/Tree: Interactive subspace visual exploration and analysis of high dimensional data. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2625–2633, 2013. doi: 10.1109/TVCG.2013.150

[76] C. Ziemkiewicz and R. Kosara. Laws of attraction: From perceptual forces to conceptual similarity. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1009–1016, 2010. doi: 10.1109/TVCG.2010.174