CS573 Spring 2022 Final Project Process Book

Truman Larson
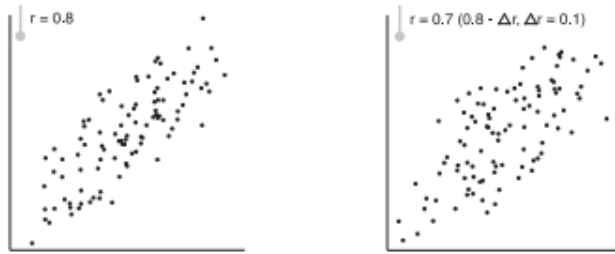

Fig. 2. An example of approaching a target correlation level from *below* in the experiments. Two side-by-side scatterplots (without any indication of actual correlation value or the regression line) are shown to the participant in the experiment. The participant chooses which of the two appears to be more correlated.

Figure 1: Methods used by Yang

Figure 2: List of conceptual visual features Yang theorized for how we perceive correlation in scatterplots.

1 Overview and Motivation

Animation in visualization is an area that has been criticized as a ineffective method of conveying information [4]. Realistically, animation is a tool just like any other, and has its good and bad use cases. One opportunity is in the area of visualization judgment, specifically in correlation. Some visual features have already been identified and evaluated [5], but there remains an opportunity to explore other types of features, specifically animated ones. The goal of this project is to take the first few steps in that direction and explore the potential and challenges of animation visualization, while also creating a tool that can assist with understanding what correlation visualization methods work well together. Eventually, this work could be adapted into an experiment, like the one shown in figure **??**, to determine the validity of animation in correlation judgment and to see specifically what methods work the best. For now, creating a robust, expandable, and intuitive tool is the priority.

2 Related Work

Figure 2 shows a list of visual figures developed by Yang et al. [5]. This work is the main basis for this project and their evaluation of these visual features proved invaluable. The main gap in the utilization of these features is the fact that they are static. Larsen et al. provides context as to how people compare patterns and suggests that mental translation may be a significant factor [1]. Thus, including animation in the list of visual features serves only to complete the list.

3 Questions

- Does animated feedback improve individuals' performance in correlation judgement?

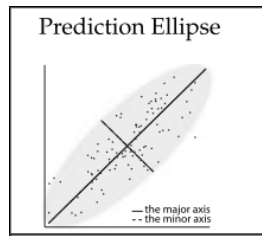   While the scope of this question is broad,

Figure 3: Ellipse visual feature example [5]

Figure 4: Bounding box visual feature example [5]

Figure 5: Distance to line visual feature example [5]

it is ultimately what this project is working towards. Overtime, this question was narrowed to the specific details of implementation and feasibility, but this remained the penultimate goal throughout.

- What animation types are best for reinforcing correlation perceptual ideas?

    As seen in Larsen et al., metal translation was a good starting point for answering this question. Again, as I moved forward with this project, it became more a question of feasibility and implementation.

- What visual features work well together?

    This is what I believe is the core question that this project specifically is trying to answer. Giving a platform to judge and play with different combinations of animation and visual features will give a platform for future work.

4   Data

While this project doesn't specifically use a data source, it is worth mentioning that all of the data used in the charts, and the charts themselves, are generated using code provided by Lane Harrison and Yiren Ding. The code provides a way to generate data that has a particular correlation coefficient. This data can easily be displayed using the chart code they provided as well.

5   Exploratory Data Analysis

Since there is no data to analyze really, this section is mainly going to be looking at the original designs used by Yang et al.   Figures 4 through 6 were the best performing figures outlined by Yang et al. Out of these features, I was personally drawn towards the ellipse as a feature due to its simplicity and visually appealing nature. The distance to line does
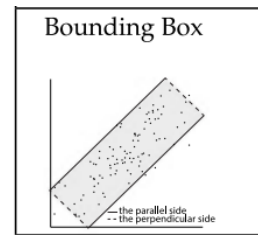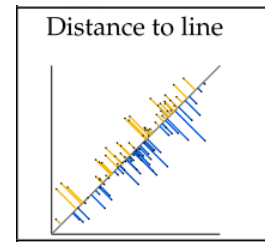
seem helpful, but would need significant efforts to declutter to make more intuitive. I see the idea behind the bounding box, but it just seems way too clunky to be useful. The convex hull was the one that confused me the most, but I can see its usefulness.

The other big aspect that these designs revealed to me was the necessity of a trendline in these types of graphs. It gives a really good reference to compare each point to and provides a bit more context that I think would be useful in any chart.

6   Design Evolution

After the initial evaluation of the designs, I designed and implemented the prediction ellipse and the bounding box with Madyson Kelly in place of my Assignment 3. The aspects of the feedback design themselves were quite simple as they were clearly defined, though the implementation was a bit less straight forward. The important part of the design was the application itself. When first designing this tool, we emphasized the importance of a few key aspects: modularity, expandability, and ease of use. The application has to be modular so that many different combinations can easily be tested and judged. A modular application would mean maximizing the reusability of the individual parts, making the level of "mix and match" that we wanted possible. Expandability is an important aspect since we initially did not know how many different types of charts we would want and what add on features we would end up needing. Making an application that was modular and expandable
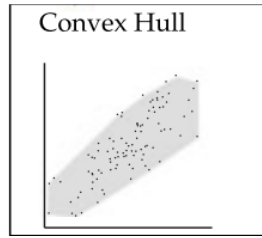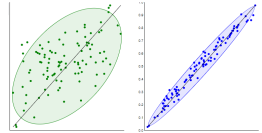
Figure 6: Convex hull visual feature example [5]

Figure 7: Ellipse visual final implementation

Figure 8: Bounding box final implementation

Figure 9: Distance to line final implementation

meant that these parts could be added on easily with minimal editing. Lastly, we wanted to emphasize the ease of use of the tool. Making it easy to use gives more motivation to try more types of combinations. If we make it ugly and unintuitive, no one would even want to use it making it useless.

Next, we evaluated what animation and add-on features would best fit the application. As mentioned before, translation of the elements is a base animated feature we wanted to try. The best way to do this, however, was not as clear. Initially, we thought to combine the two charts and simply change the color so that the user could compare one color to the other. While this does work, it still is very cluttered and hard to read. Later in the project, I got some better advice to make the comparison an active process, having the user click to highlight one chart or the other. This proved to be much more helpful overall.

Moving through the experiment, I found that most people I asked to check out my project would tilt their head to view the comparison. This prompted similar research into rotation, but none such research was easily found. However, rotating the charts such the theoretical trendline was horizontal proved to be much easier to read and compare, especially when the two charts were combined.

This project does deviate from the proposal greatly. This was mainly an issue of scope since the proposal describes a full experiment that would not have been possible. The scope was shrunk to just lay the groundwork for an experiment of that nature.
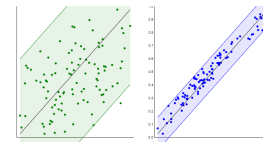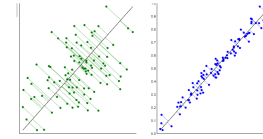
7 Implementation

For the final charts shown in figures 7 through 10, the implementation was mostly hindered by technical details rather than design. Much of the time was spent figuring out exactly how to get the features in the correct spot such that the visualization was accurate. Some of the visualization had algorithms provided to us. The ellipse algorithm was given by Lane Harrison provided in turn by the authors of Yang et al. Additionally, the convex hull algorithm found online and provided by Project Nayuki (https://www.nayuki.io/page/convex-hull-algorithm).

A large part of the implementation was also spent realizing the design principles planned out in the previous section. Each chart is in its own file, and each add on also resides in its own file. There is a simple method for passing information about the underlying charts between each of the functions.

Most of the charts also have some element of animation build in within them. This was mainly for aestetic purposes. The add-on, like the trendline and axis fade, are also animated for this reason.

The core animations, shown in their completion in figures 11 and 12 were relatively trivial to implement using d3 animations. As figure 12 shows, having both of the animations together results in a clear and easy comparison.

One of the final core features is the toggle feature. This isn't very obvious at first, but one the user is used to the website, it becomes a valueable tool for decluttering the graphs after merging. This is on by default whenever the graphs are merged together and clicking them will toggle which graph is highlighted by changing the color and opacity.
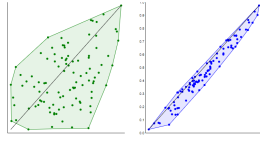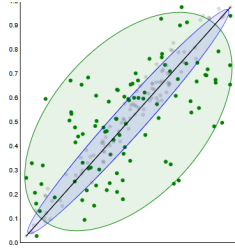
Figure 10: Convex hull final implementation

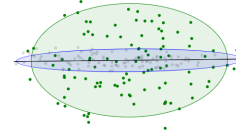Figure 11: Merge final implementation

Figure 12: Merge and rotate final implementation

8 Evaluation

References

[1] A. Larsen and C. Bundesen. Effects of spatial separation in visual pattern matching: Evidence on the role of mental translation. *Journal of Experimental Psychology: Human Perception and Performance*, 24(3):719, 1998.

[2] J. Li, J.-B. Martens, and J. J. van Wijk. Judging Correlation from Scatterplots and Parallel Coordinate Plots. *IVI*, 2010.

[3] R. A. Rensink. The nature of correlation perception in scatterplots. *Psychonomic Bulletin & Review*, 24(3):776–797, Jun 2017.

[4] G. Robertson, R. Fernandez, D. Fisher, B. Lee, and J. Stasko. Effectiveness of animation in trend visualization. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1325–1332, 2008.

[5] F. Yang, L. T. Harrison, R. A. Rensink, S. L. Franconeri, and R. Chang. Correlation judgment and visualization features: A comparative study. *IEEE Transactions on Visualization & Computer Graphics*, 25(03):1474–1488, mar 2019.
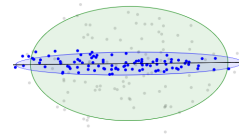
Figure 13: Toggle declutter final implementation