Jason Davies’ Example

Jason’s parallel coordinates is fairly effective when it comes to interaction. I like the way creating filter boxes and reordering the axes. The filter result can be quickly displayed and the interactive functions give very fast performance. Yet for visual pleasing reason, I think the data can be drawn through smooth curves rather than jointed lines. Another thing that maybe necessary is allowing for more data mining functions, like clustering, such that the data set can be thorough explored.

Design

The major components and functions of the visualization was roughly designed in the sketch and more details were added in the program.

a) Layout

The encodings and interactions with the parallel coordinates and clustering are the major components in this visualization. I also envisioned cluster centrals for clustering, and an interactive color legend. Thus the view is split up into four panels, as shown below: color palette, cluster legend, parallel coordinates, toggle buttons of clusters and centrals.
The parallel coordinates has occupied the largest area of the view, which is located in the middle. The color palette goes across the top of the view. The cluster legend is between the color palette and parallel coordinates. In the bottom are the toggle buttons that control the displays of the clusters and centrals.

b) Parallel coordinates: Jason Davies’ example

When designing the interactions with the parallel coordinates, I focused on a clean interface and expected the users be able to explore the data easily and effectively.

For the basic elements, like axes, data, title, labels, tick marks, etc, I envisioned a intuitive plot that is similar to Jason’s example, but in this visualization the data would be plotted through smooth curves.

The following is the sketch that contains the primary interactions:
Parallel Coords: Jason Davies' Example

1. Invert the axes.

When the cursor hovers over the bottom of an axis, display the "invert sign" — "\(^{\wedge}\)" and change the cursor's shape as a hand.

2. Reorder the axes.

Do it as the Jason Davies' Example does.
3. Filter the data across multi-attributes.

3.1. Create, move and change the filter boxes.

Drag the mouse to create a filter box, release the mouse to finish.

Hover the mouse over the area of a filter box and drag to move.

Hover the mouse over the top or bottom of a filter box and drag to change the size of the filter box.

3.2. Cursor shapes for the operations:
- Create: cross
- Resize: I
- Move: e
For inverting the axes, when the mouse hovers over the bottom of an axis, the program will display the "invert sign" ("^") and change the cursor's shape to a hand.

The reorder and filter operations are also referenced from Jason's example. As mentioned in the beginning of the report, Jason has done a good job of creating, move and change the filter boxes. Thus I borrowed that manners into my design.

Transparency is a nice way to demonstrate information with a low value and show unimportant data by fading the corresponding colors. In this visualization I planned to utilize transparency to deal with overlapped lines and filtered data:

c) Clustering

Considering the nature of clustering, I think the centrals of clustering are essential to illuminate both the trends and the correlations of data. Thus I put two toggle buttons in the bottom to control the displays of clusters and centrals. The last thing is the interactive color legend. It would be located at the top of the
visualization, so that users can flexibly color any cluster.

Clustering

1. Assign each cluster a default unique color.
   - Show all data (clusters turned on), with or without centers.
   - Show all data, with centers.
   - Just show centers.

2. Show a certain cluster.
   - Click a cluster to see the correlated data.
   - Hover the mouse over a cluster to see the correlated data.
I ended up with implementing a clear interface for the basic visualization and all the interactions. cars.tsv is employed for Basic Visualization, Interactivity and Clustering.

**Basic Visualization**

I highlighted the core panel that holds the parallel coordinates, for the purpose of facilitating the focus on major information. Here is the static parallel coordinates plot:
Interactivity

a) Filter the data across multiple attributes

There are three features in the filter function:

- Create a filter box by dragging the mouse along the axis and releasing the mouse to finish.
- Move a filter box by hovering the mouse over the box and drag the mouse to move.
- Change the size of a filter box by hovering the mouse over the bottom or top of the box and drag to resize.

Create a filter box at “Mpg”: 
Create the second filter box at “Origin”:

Change the size of the filter box at “Mpg”: 
Move the filter box at “Mpg”:

b) Reorder / rearrange the axes

To reorder the axes, users need to hover the mouse over the titles. When the cursor changes to a CROSS from ARROW, drag the mouse to move an axis to anywhere. After releasing the mouse, the dragged axis will be automatically placed at the middle of the two axes in its sides.

Continue the operation from last screen shot: move the axis of “Weight”: 
After exchanging the axes of “Weight” and “Acceleration”, “Origin” and “Model.year”:

c) Invert the axes

When the mouse hovers over an axis, an “arrow” shaped as “∧” will appear and the cursor will change to a HAND. As shown below, if we want to invert the axis of “Horsepower”, hover the mouse over the bottom then the arrow appears:
Clustering

a) The k-means clustering algorithm

I implemented a k-means algorithm for the clustering, which supports labeling the data into different clusters, computing the centrals and returning the accuracy. For the clustering of the car data, I tried...
different cluster number for the training and learning. It turned out that the data can be clustered into 5 clusters with a relative high accuracy of 74%.

b) Clusters and centrals

There are two toggle buttons in the bottom of the visualization, click any button to turn on the clusters or the centrals, or simply hover the mouse over the buttons to view the clustering.

If we turn on the clusters, the data from last screenshot will be colored as the clusters they belong to. The cluster legend is above the parallel coordinates:

The whole clusters are displayed as below:
Turn on the centrals, the five highlighted curves are the centrals and the data are faded:

To purely observe the centrals, turn off the clusters:
To view any single cluster, click the corresponding cluster in the legend area:
An alternative manner of viewing the clusters is hovering the mouse over the cluster legend:
c) Interactive color legend

The color legend is located in the top of this visualization. I picked the standard 216 color palette for this feature:

To change the color of a certain cluster, select the cluster and pick one color from the palette:
Data Exploration

a) Cars

Turn on the clustering:
Filter the data to eliminate small proportion of curves that are disturbing the view:

Turn on the centrals to see how the clusters go across the axes:
Reorder the axes to get the strongly correlated attributes close to each other:

It is clear to see that $Mpg$, $Displacement$, $Horsepower$ and $Weight$ are strongly correlated.

b) Flights

I explored the departure and arriving information of the most US flights in 2009. The data were ideally clustered into 4 clusters with the accuracy of 85%.
The first thing I did was assigning the clusters with different colors:

After reordering, inverting axes, the strong correlation between CRS_DEP_TIME, DEP_TIME, CRS_ARR_TIME and ARR_TIME is obtained:

Critique

From my perspective, an ideal parallel coordinates should be able to allow as much necessary interaction functions as possible. Because the intuitive usage of parallel coordinates is displaying information with multiple
attributes, major information need be explored through interactions. Without those interactions, most features would be hard to be found.

Here I briefly summarize the pros and cons of the visualization:

This parallel coordinates system allows users to flexibly explore quantitative data with the supported features of reordering axes, inverting axes, filtering data across multiple axes and clustering the data. Detailed encodings are also implemented to better convey the information: changing highlighting (the transparency utilization in filtering, clustering), changing spatial ordering (reordering, inverting), linked highlighting (viewing single clustering, picking color through the color palette).

The main drawback of this system is the insensitive rendering speed for large data set. When there are very large data to be shown, I found some interactions become pretty slow, like rearranging. Another limitation is the limited usage for quantitative data. Other kinds of data are not valid in this system.

**Improvements**

Regarding to the critique above, the improvements can be made by speeding up the interactive operations, displaying non-quantitative data in charts or other kinds of visualizations through linked views with the parallel coordinates.