# Implementing Data Visualizations with a brush-and-link interaction. Report for the HPDAV Second Assignment

Luca Guffanti

November 25, 2024

## 1 Introduction

The objective of the project is to construct an effective visualization to present bike sharing data, registered in the city of Seoul. Each data point contains information about the number of bikes rented at a given day and time, along with various environmental parameters. More specifically, the `RentedBikeCount, Hour, Temperature, Humidity, WindSpeed, Visibility, DewPointTemperature, SolarRadiation, Rainfall, Snowfall` fields are numeric; `Date` is treated numerically (through some preprocessing), while the `Seasons, Holiday, FunctioningDay` attributes are considered as nominal. The visualization, of which a picture in Figure 1, is implemented in `react.js` and `d3.js`, adopting a **Model-View-Controller** design approach, powered by `redux.js`.

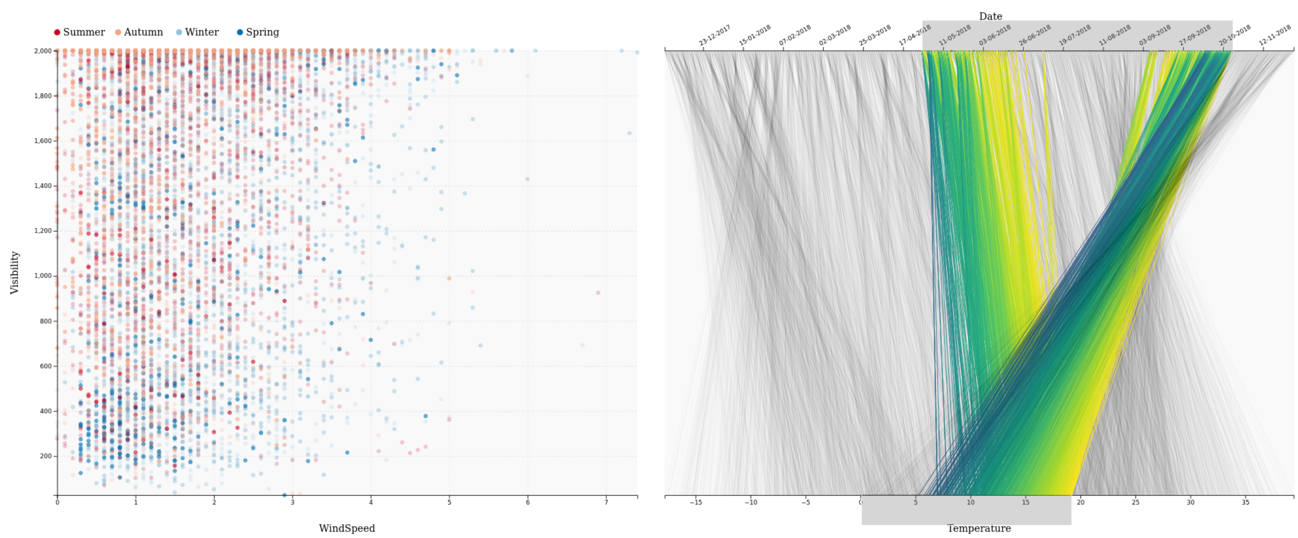


Figure 1: Visualizing bike sharing data with a scatter plot and parallel coordinates plot with two coordinates.

## 2 Design Choices

### 2.1 Scatter plot

Position is one of the most effective visual variables when considering numerical attributes. Thus, a scatter plot is a natural choice for the visualization of the many numerical properties of the data, as it exhibits the relations between two variables. In the scatter plot the user has the opportunity to choose between a pair of numerical attributes (plus `Date`, which is treated numerically) to construct axes with the dimensions of interest. Additionally, the user can select a categorical attribute that defines the coloring and sizes of the dots (as shown in Table 1), because it was noted that, due to the high number of data points, using a number of different symbols and visual variables to encode categorical information would overly clutter the visualization, especially in already dense clusters, leading to readability problems. Colors have been chosen to highlight differences, showcasing, especially for `Holiday` and `FunctioningDay`, the outliers. Along this line, effort was put into creating a dynamic legend that would be updated with the change of the chosen categorical.

The scatter plot supports 2D brushing, implemented with the `d3.brush()` component. As shown in Figure 2, brushed data points are increased in opacity.

| Holiday | Color |
|---|---|
| `Holiday` | |
| `No Holiday` | |

| Season | Color |
|---|---|
| `Spring` | |
| `Summer` | |
| `Autumn` | |
| `Winter` | |

| FunctioningDay | Color |
|---|---|
| `Yes` | |
| `No` | |

Table 1: Left: encoding for the `Holiday` attribute. Center: encoding for the `Season` attribute. Right: encoding for the `FunctioningDay` attribute.

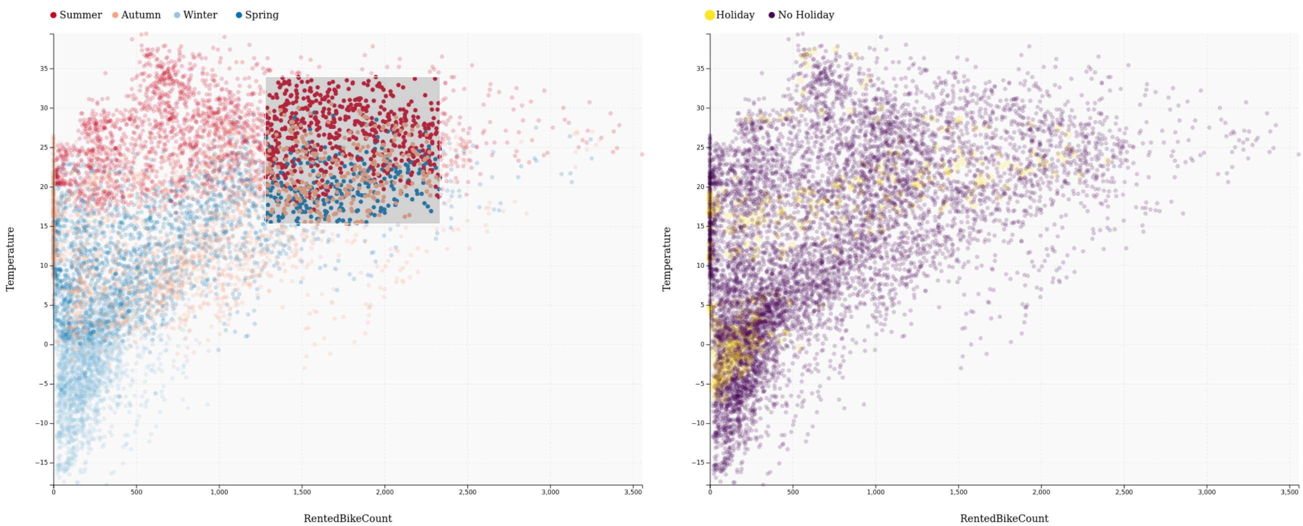


Figure 2: Left: 2D brushing on the scatterplot. Right: `Holiday` attribute chosen as color encoding.

## 2.2 Parallel Coordinates plot

This visualization was chosen for its effectiveness in qualitatively illustrating proportionality relationships and tendencies within the data, already partially understandable through the scatter plot. Parallel coordinates offer interesting insights especially when different groups of data points exhibit local patterns and trends. As for the scatter plot, the user has complete control over the numeric attributes that are displayed on the axes, which can also be inverted. This second plot supports 1D and 2D brushing, the latter being implemented as the combination of two 1D brushes, one for each axis. When an interval is brushed, the visualization is updated to highlight and color the selection. Based on the brushed axis/axes, a color scale is dynamically mapped on the selected domain: if the upper axis is the only brushed one, than the color scale domain is the upper axis selection; if the lower axis is the only brushed, then the domain of the color scale is the lower axis selection. If both axes are brushed, then the domain of the color scale is the narrowest selection. The color scale uses the Viridis color map, which is established in the literature for its effectiveness. This design choice (Figure 3) proves useful in highlighting tendencies and contrasts within small portion of the data, ordered along a specific dimension.

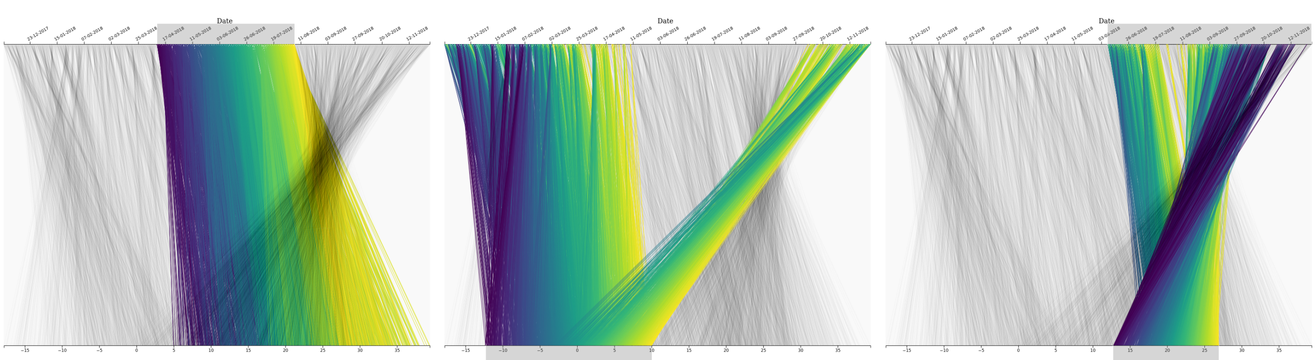


Figure 3: Left: 1D Brushing on the upper axis. Middle: 1D Brushing on the lower axis. Right: 2D brushing. The lower selection is the narrowest and it controls the color scale domain.

## 2.3 Inter-plot brush-and-link

As shown in the previous sections, both plots collaborate to provide the user a comprehensive overview of the dataset, also through interaction by brushing. Additionally, the two plots are linked, meaning that brushed selections on either the scatter plot or the parallel coordinates plot are reflected on the other visualization. Selecting data points in the scatter plot reveals the same in the parallel coordinate plot (with a static color scale mapped on the lower axis domain) and vice-versa, as shown by Figure 4.

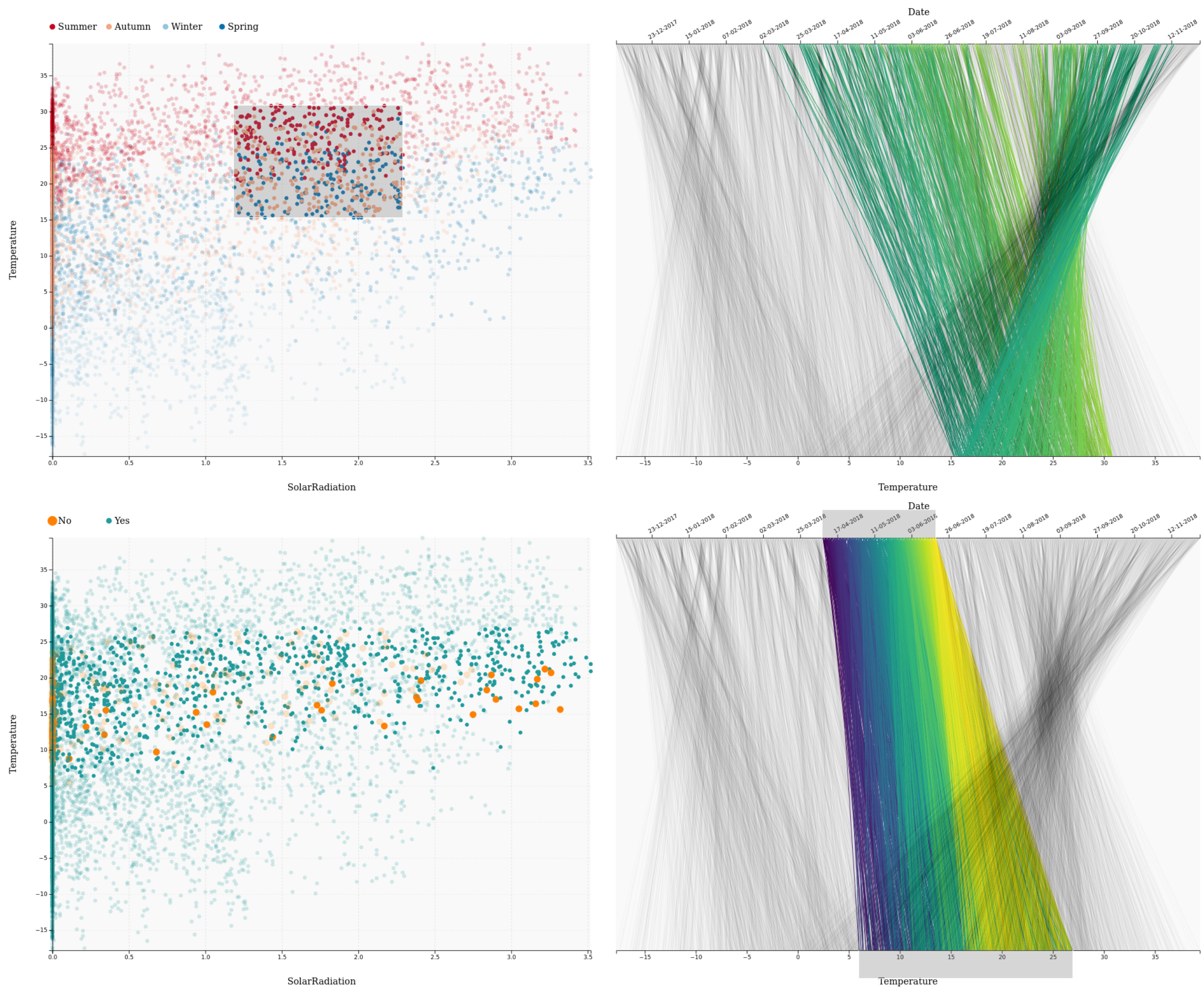


Figure 4: Top: A selection in the scatter plot is linked to the parallel coordinate plot. Bottom: A selection in the parallel coordinate plot is linked to the scatter plot.

# 3 Final Considerations

The constructed visualization is effective in showing trends present in the data: the parallel plot helps displaying local tendencies while the scatter plot shows clustering, with the brush-and-link interaction allowing for a finer analysis of the data points. The opportunity of selecting four different axes (two for the scatter plot and two for the parallel plot) and a categorical color encoding is beneficial for the versatility of the visualization. Nonetheless, selecting the same attributes on both plots can be helpful for cross validating predictions.

Notably, this visualization cannot substitute an in-depth statistical analysis because the sheer amount of data makes reasoning on the dataset mostly qualitative. Similarly, it also lacks the effectiveness of aggregated visualizations. Plotting a large amount of data contributes to visual cluttering, and the limited screen space obliges the use of smaller dots. Additionally, lines in the parallel plot must be very opaque as they tend to cross each other often. In conclusion, this visualization proves effective in providing a clearer understanding of the data. However, its true potential is realized when it is integrated into a broader context, where quantitative characteristics and aggregated results are also highlighted and explored.