Information Visualization Based On Various Tools

5140309198 YangJiacheng

Abstract

Information visualization?is the study of (interactive) visual representations of abstract data to reinforce human cognition. The abstract data include both numerical and non-numerical data, such as text and geographic information.
In this project, the output of TSNE will be presented on a website page, which is saved in several txt documents. The data consists 500*2500*2 figures which stand by the location of the points. I chose d3.js and plotly.js to achieve the visualization.

Keywords: information visualization,web programming JavaScript,API

1 d3.js

d3.js (or just d3 for Data-Driven Documents) is a JavaScript library for producing dynamic, interactive data visualizations in web browsers. It makes use of the widely implemented SVG, HTML5, and CSS standards.
d3.js API contains several hundred functions, and they can be grouped into following logical units:Selections,Transitions,Arrays and etc.
Embedded within an HTML webpage, the JavaScript d3.js library uses pre-built JavaScript functions to select elements, create SVG objects, style them, or add transitions, dynamic effects or tooltips to them. These objects can also be widely styled using CSS. Large datasets can be easily bound to SVG objects using simple d3.js functions to generate rich text/graphic charts and diagrams.
The central principle of d3.js design is to enable the programmer to first use a CSS-style selector to select a given set of Document Object Model (DOM) nodes, then use operators to manipulate them in a similar manner to jQuery, which is called chain syntax.
Chain syntax is achieved by several functions in the correct order. Most of the functions will return a selection. the output of a function in the chain is requested to be the same as the input of the next function, or the chain will break. For example:

$$d3.select(''body'').append(''p'').text(''Newparagraph!''); \tag{1}$$

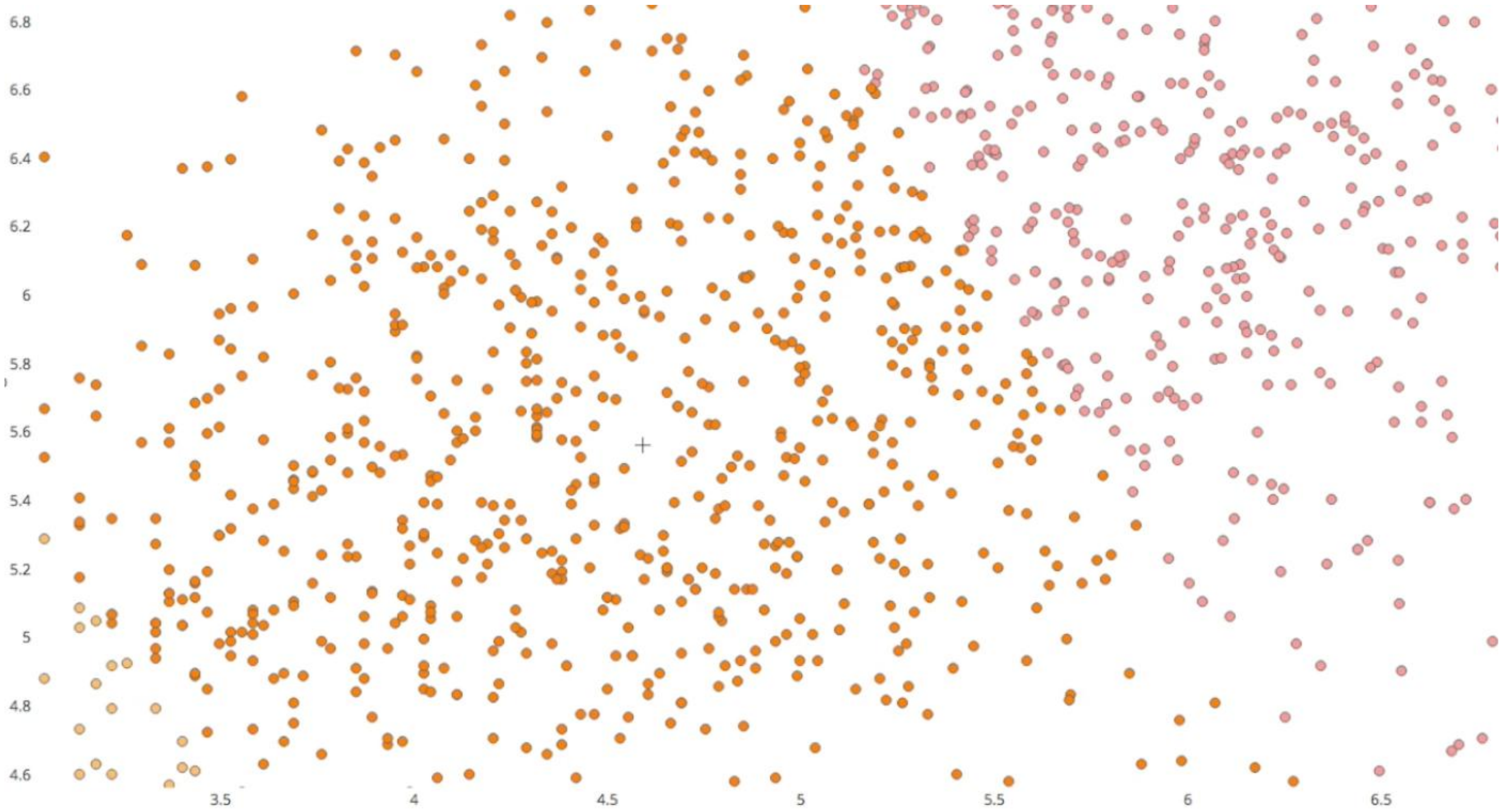
The construction of this code is:
CCS selection – .select(”body”) – DOM element – .append(”p”)
– New body ”p” – text(”New paragraph!”) – The text
Dimple is an object-oriented API for business analytics powered by d3.
The aim of dimple is to open up the power and flexibility of d3 to analysts. It aims to give a gentle learning curve and minimal code to achieve something productive. It also exposes the d3 objects so you can pick them up and run to create some really cool stuff.

2 plotly.js

Plotly, also known by its URL, Plot.ly, is an online data analytics and visualization tool headquartered in Montreal, Quebec. Plotly provides online graphing, analytics, and statistics tools for individuals and collaboration, as well as scientific graphing libraries for Python, R, MATLAB, Perl, Julia, Arduino, and REST.
I tried plotly.js, an open source JavaScript library for creating graphs and dashboards.
plotly.js is a high-level, declarative charting library. plotly.js ships with over 20 chart types, including scientific charts, 3D graphs, statistical charts, SVG maps, financial charts, and more.

3 comparison

When I was learning these two API and to use them in the information visualization, I noticed lots of beautiful examples.
Most of the examples of dimple.js, the data is formed by a csv chart with several rows and several columns, which is commonly seen in business and report of a company. In this case, we can easily get different data of one element such as the price and sales. On the other hand, the data used in plotly.js is divided into some sets which are called traces, so that if we want to show different types of data in one graph and avoid mistaking them, plotly.js may be a better choice.
Also we can find that both tools provide excellent interface for users when you need more details of the data, by putting your mouse on the nodes or the axis. However, I think the plotly.js do better at zooming of the chart.
In addition, there are many interesting functions used in 3D-chart in plotly.js.
In this project, since the data of our data is simple, and we only need to present scatter points. I prefer to use plotly.js to achieve the visualization.
The final result is performed here:


6.8
6.6
6.4
6.2
6
5.8
5.6
5.4
5.2
5
4.8
4.6
3.5
4
4.5
5
5.5
6
6.5