

M4: Final Submission

Nicole Cherches
Matr. Nr.: 01506832

Alexander Gelb
Matr. Nr.: 01268620

Benjamin Neckam
Matr. Nr.: 01301917

Axynia Tokareva
Matr. Nr.: 01268620

ABSTRACT

Some blabla here

Index Terms: Visualization—Visualization techniques; GAIA—ESA

1 MOTIVATION

On 19th December 2013 the "European Space Agency", ESA, launched the "Gaia"-satellite to gather more information about our galaxy and create a three dimensional map out of it. Since then data like velocities, positions, errors, parallax and many more were collected of about 2 million stars. All together each star has 50 features in the data set which means the amount of entries is extremely high. But working with such a huge dataset is not easy and most of the time not necessary because scientists only want to focus on a specific area of the galaxy. Therefore they only use subsets of the data. Now the idea was what happens if we take the whole data set and don't look into detail but see the "big picture" and try to find patterns, correlations, outliers or other interesting things.

This could be useful for scientists who are not sure which part of the data they want to explore and for example can focus only on the anomalies or other fascinating sections in the data set.

2 RELATED WORK

What we first did, was searching for similar problems and finding out, what other visualizations looked like in this area. Most of the space visualizations we found online, were in 3d. Examples of them are the ESA Star Mapper [1] or 100000 stars [2]. They both are 3d interactive simulations of the stellar neighbourhood. We also found a 2d-visualization from the ESA, which visualizes the stars from the Gaia dataset [3]. But we couldn't find anything that has to do with correlations or patterns. All the existing visualizations were about showing the stars themselves with their exact location. So we started prototyping and in the end we even used some of our first basic ideas in the end product of our system. Then we tried to find existing visualization systems. In the VIS course we already learned about Tableau [4], which allows the user to import Data and create visualization plots based on it. And we also found out about a software named Glue [5], which helps to create plots and analyze relationships between different data. So we basically tried to create a mixture of both tools and started designing our own system.

Above is a picture, that shows how we imagined the visualization system at the beginning. As we will see later in the end product, our Lo-Fi Prototype looks similar to our visualization system. We kept the sidebar, where the user has basic information about the data and can choose options on how the data will be presented. Also the diagrams and plots appear all on one screen so the user can see how they correlate.

3 APPROACH

When we first thought about our visualization design, we tried to think about our user. The system should help to see "the big picture"

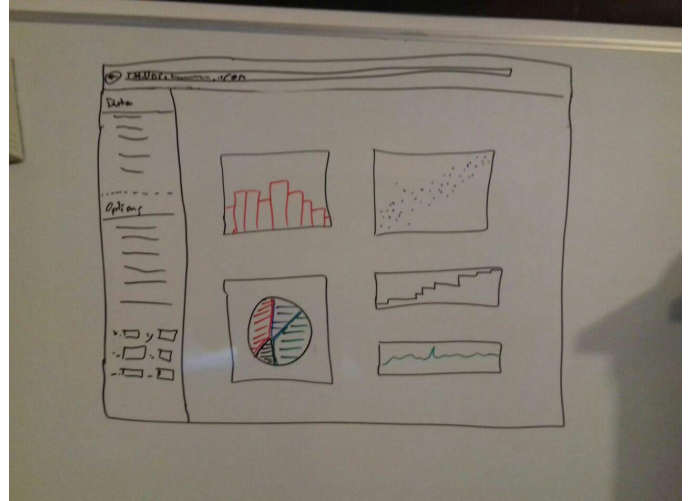


Figure 1: Lo-Fi Prototype of our user interface

of a dataset and not only details. So, we thought about plots that could help us to realize that.

What we decided to do, was a Scatterplot Matrix: It really fits into our project, because we can see a whole grid filled with Scatterplots. This way, we can see all Correlations between selected columns of the Dataset. The user has to choose a set of columns, he wants to analyse in the sidebar. The scatterplot has also histograms in the diagonal, so we can see the patterns even better.

Our system can also be used to compute histograms separately. Here the function creates bins for a specific column and shows how many entries are in every bin. While hovering over the bars from our plot, we can also see the exact number of those entries. The user can also set the bin size manually.

Then the system also provides a function to compute a Correlation coefficient. For this, the user has also to select a set of columns. For these columns the function computes and displays the correlation coefficient. So it is easy to see how the different columns of a set are connected and if the user needs to analyse it further.

The last part of the system is the PCA. Blabla ber das PCA.

4 IMPLEMENTATION

The whole program was written in JavaScript combined with CSS. Therefore it works on every web browser. For the front-end we decided to use the toolkit "Bootstrap" [6] which provides different templates so we needn't to spend too much time for the user-interface. To visualize the data the JavaScript library "D3" [7] was used which offers the ability to let the user interact with the data itself.

We encountered many problems which challenged us to implement things properly. One of the problems was to filter the data that all plots can be realized. We figured out that we just can't delete all the entries which contain a NULL or any other non valid values because this would reduce the data set to almost zero entries. Therefore we went through the documentation of the dataset and tried to figure out which features of the stars can be excluded and

which not. Unfortunately we can not filter out all unnecessary values at the beginning, we have to do a "pre-filtering" to remove all non integer values and afterwards almost every function has to do their own filtering to adapt the data to their specific task.

5 RESULTS

We started with making a side bar, which shows a few details about the dataset and allows the user to make some specific settings. On top, we can see the Dataset name, rows and columns of our dataset. Firstly, the user can choose which kind of plot he wants to produce in the Options window. "Data Value" is a drop down, in which the user can choose exactly one column, which is what we need for the histogram. If he chooses the Scatterplot, he needs to choose more columns in "Multiple Data". After that, he clicks the "Plot" Button and the chosen diagram will be visible on the display. With the "Remove" Button, he can clear the page.

6 DISCUSSION

6.1 Lessons learned

6.1.1 Benjamin Neckam

There were two things I will take out of this project.

First refers to the implementation and the chosen programming language. Basically I was not very used to JavaScript, to be more precisely I had no experience with it, but I was really impressed by "D3". The opportunities you have to visualize data is breathtaking and allows the programmer to choose over a wide variety of plot types. The only disadvantage that really falls into weight was the performance. If the dataset is bigger than, let's say, 100.000 entries it takes some time to calculate and plot the data and in our case the file contains more than ten times more entries. But all in all it was a very interesting and in my opinion very important for my future to work with JavaScript and "D3" and get deeper into this part of front end programming.

The second part is about the collaboration between the project group of the class and research group of another university. Until now all the projects at the university were from the teachers of the faculty itself but in this case the task was from the "Universitätssternwarte Wien". This means we had to arrange meetings with the project leader and a student which also is a member of the research group and talk about the task in detail and what the aim of this project is. It was fun and interesting to meet all these people but I recognized that even if there are not too many people involved it can easily happen that confusion arises. Like in our case, we started to focus on the wrong details and parts and had to "restart" our project. Therefore I would say communication is the most important part in a project, especially if it is a collaboration of two or more groups, I guess this is the biggest lesson I learned from the project.

7 SEPARATION OF TASKS

REFERENCES

- [1] Jan Willem Tulp, Jos de Bruijne, Karen O'Flaherty, and Claudia Mignone. A visualisation based on data from the european space agency's hip-parcos star mapper. http://sci.esa.int/star_mapper. Accessed: 17.01.2018.
- [2] Google Data Arts Team. Interactive visualization of the stellar neighborhood. <http://stars.chromeexperiments.com/>. Accessed: 17.01.2018.
- [3] Andr Moitinho de Almeida, Hlder Savietto, Carlos Barata, Alberto Krone-Martins, Mrcia Barros, Antnio Falco, and Tiago Fernandes. Interactive visual exploration environment for the gaia archive. <https://gea.esac.esa.int/visualization/index.html>. Accessed: 17.01.2018.
- [4] Tableau Software. Interactive data visualization product. <https://www.tableau.com/de-de>. Accessed: 17.01.2018.

- [5] Chris Beaumont, Thomas Robitaille, and Michelle Borkin. Python library to explore relationships within and among related datasets. <http://www.glueviz.org/en/stable/>. Accessed: 17.01.2018.
- [6] Twitter Inc. Open source toolkit for developing with html, css and js. <https://getbootstrap.com/>. Accessed: 18.01.2018.
- [7] Mike Bostock. Javascript library for visualizing data with html, svg, and css. <https://d3js.org/>. Accessed: 18.01.2018.